REPORT DOCUMENTATION PAGE

Form Approved
OMB NO. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	Sept 96	Final 12 Sep 93 - 30 Aug 96

4. TITLE AND SUBTITLE
FLIR/LADAR Fusion for Target Identification

5. FUNDING NUMBERS
DAAH04-93-C-0051

6. AUTHOR(S)
Thomas Jenks

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(ES)
Rockwell International Corporation
Anaheim, CA 92806-2869

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Research Office
P.O. Box 12211
Research Triangle Park,, NC 27709-2211

10. SPONSORING / MONITORING AGENCY REPORT NUMBER
ARO 32366.1-MA

11. SUPPLEMENTARY NOTES
The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy or decision, unless so designated by other documentation.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12 b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Rockwell's major successes during the contract period are the following items.

- **Developed A New FLIR/LADAR ATD/I Framework:** A foundation for an innovative ATD/R/I system was developed for others to use. The new approach incorporates state of the art techniques such as FLIR/LADAR feature level fusion with clutter suppression and hierarchical classification algorithms.
- **Developed FLIR-Based Background Suppression Software:** A software package for integration into the UGV was developed and delivered to Lockheed Martin in Denver during the month of December, 1995.

14. SUBJECT TERMS	15. NUMBER IF PAGES
	16. PRICE CODE

17. SECURITY CLASSIFICATION OR REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

C94-14/034

FLIR/LADAR Fusion for Target Identification

Final Report

Defense Advanced Research Projects Agency
3701 North Fairfax Drive
Arlington, VA 22203-1714

Contract DAAH-04–93–C-0051

Autonetics & Missile Systems Division
Rockwell International Corporation
3370 Miraloma Avenue
Anaheim, CA 92803-3105

August 30, 1996

Foreword

This final report is provided in accordance with item number F.2.e of the FLIR/LADAR Fusion For Target Identification (FLFTI) contract (number DAAH-04–93–C-0051). The research was conducted in cooperation with the Unmanned Ground Vehicle (UGV) program under the sponsorship of the Defense Advanced Research Projects Agency (DARPA).

Contents

1 Summary

Rockwell International's objective was to develop a robust and state of the art FLIR/LADAR target detection and identification system for the reconnaissance, surveillance, and target acquisition program. The algorithm suite was to be integrated into the Unmanned Ground Vehicle (UGV) platform. But due to 1) program changes such as the late availability of the LADAR sensor unit and 2) funding restrictions in calendar year 1996, the primary goal was able to be addressed in this contract. This report describes the work accomplished towards the objective; documenting the major successes and conclusions that were obtained in the process.

Rockwell's major successes during the contract period are the following items.

- **Developed A New FLIR/LADAR ATD/I Framework:** A foundation for an innovative ATD/R/I system was developed for others to use. The new approach incorporates state of the art techniques such as FLIR/LADAR feature level fusion with clutter suppression and hierarchical classification algorithms.
- **Developed FLIR-Based Background Suppression Software:** A software package for integration into the UGV was developed and delivered to Lockheed Martin in Denver during the month of December, 1995. It incorporated many of the FLIR-based background suppression ideas in this report.

During the process of planning, research, and the development of algorithms towards the objective, the following conclusions were reached:

- To obtain high detection, low false-alarm rates, and robust identification of targets, one must deal with background suppression at the onset. It must be integrated into any planned ATD/R/I system. Ground rules must be established on the difference between targets and clutter objects.
- Humans recognize objects first as categorical levels. Hierarchical classification techniques, that were developed under this contract, learn categorically. The approach shows much promise in advancing ATD/I technology.
- Fusing FLIR and LADAR data into a common feature vector as discussed in this report is a powerful method in exploiting the input data.

2 Introduction

The FLIR/LADAR Fusion For Target Identification (FLFTI) program was designed to aid the Unmanned Ground Vehicle (UGV) Reconnaissance, Surveillance, and Target Acquisition (RSTA) program. The program's goal was to improve the state of the art in Automatic Target Detection and Identification (ATD/I). Beginning on September 12, 1993, it covered a 36-month period. The contract had Option I renewed for calendar year 1995. In addition, two no-cost extensions were granted through August, 1996.

This report describes the work accomplished between September 12, 1993 through August 30, 1996 with emphasis on results after Demo C and Demo II.

2.1 Overall Objective

Rockwell International's objective was to develop a high performance FLIR/LADAR sensor algorithm suite for target identification that advances the state of the art in image understanding within a Surrogate Semi-autonomous Vehicle (SSV) and RSTA environment.

The program's objective was to develop and demonstrate

- adaptive background suppression (both FLIR-based and LADAR-based),
- environmental characterization/prediction, and
- an enhanced, FLIR/LADAR ATD/I system with adaptive, model-based capability (with further capacity to perform FLIR/range-to-target identification).

As shown in Figure 1, a three-pronged approach that incorporated (1) adaptive background suppression, (2) environmental prediction, and (3) adaptive model-based techniques was developed to implement the above objective. The aim was to modify the existent

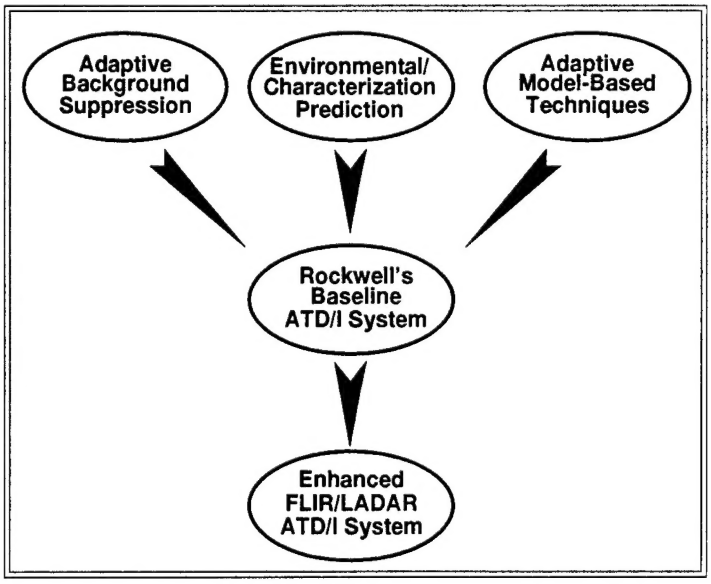

Figure 1. An baseline plan called for adaptive background suppression, environmental prediction, and adaptive model-based techniques to modify Rockwell's baseline system in order to arrive at the desired version.

Rockwell (baseline) ATD/I algorithm suite. The outcome was to be an enhanced ATD/I version that would achieve a high degree of compatibility within the RSTA/UGV framework. But funding restrictions for calendar year 1996 and the late availability of the government furnished LADAR sensor unit (beyond Demo II) curtailed the effort.

Figure 2 describes the changes to the original approach. The diagram shows that the background suppression module became the core for the enhanced ATD/I system (in generic terms, background suppression algorithms reduce the interfering effects of the surrounding target region during detection and identification). The move was wise; since, the feature extraction/classification paradigm was intended to be used for the overall system from the beginning of the program. The background suppression module (FLIR and LADAR versions) became the **testing** ground for core algorithms that were to be used again for the enhanced ATD/I system. In addition, capabilities from earlier baseline ATD/I work plus new model-based techniques were channeled into the background suppression algorithm suite.

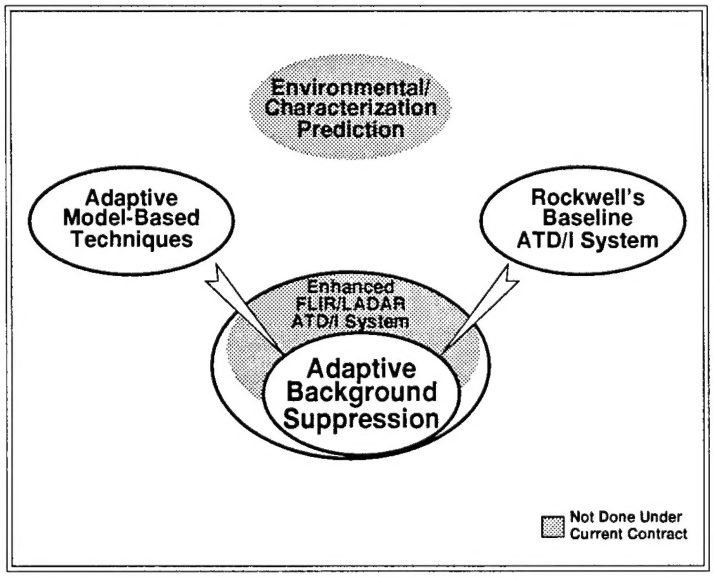

Figure 2. Due to program restrictions for calendar year 1996 and the late availability of the LADAR sensor, only a part of the enhanced system was completed. With inputs from the Rockwell baseline and adaptive model-based techniques module, the adaptive background suppression module became the core of the new system. The environmental prediction module was omitted altogether.

2.2 Demo II Objective

Rockwell's tasks for Demo II were: (1) to have the FLIR-only, background suppression algorithm integrated into the UGV environment for real-time use; and, (2) to implement an enhanced, FLIR-based or FLIR/LADAR ATD/I system as a laboratory demonstration. A version of the background suppression software that conformed to the Interface Control Document (ICD) specifications was delivered to Lockheed Martin in December 1995 for incorporation into their SSV's. But due to funding restrictions, no laboratory demo for the second task was possible.

3 Methodology

From the outset, the aim was to build, in stages, on the successes of (1) previous work performed during the contract; and, (2) software gleaned from earlier in-house algorithms (see [Roc94] and [GW94]). While working through the limitations mentioned in Section 2, the goal now became to use the background suppression module as the framework with which to develop the enhanced ATD/I system. The underlying motivation was to advance the state of the art in ATD/I systems by the following key idea.

> *In order to detect and identify tactical military targets, new systems must come on line that are able to process data quickly and adapt to ever-changing environments. One manner in which this goal can be accomplished, is by converting and condensing the input imagery efficiently so as to make the process invariant to changes in translation, rotation, scaling, and other types of deformation. Powerful categorization techniques can be applied next to the data so that the system is able to adaptively cope more effectively under high background clutter and target occlusion/articulation.*

A feature extraction/classification paradigm was used to implement the above idea (this effort is shown pictorially in Figure 3). The design was general enough so that it could be adapted for FLIR-based, LADAR-based, or fused FLIR/LADAR target detection and identification systems. The plan was to take the input imagery, after selecting likely target sub-regions, convert them into a designated transform space (*e.g.*, log-polar or 3D Hough) for invariant purposes, then perform classification on a condensed version of the set. After appropriate training off-line with a representative feature vector database, the classification process would then detect targets in clutter for the background suppression algorithm or recognize sub-parts towards target identification in the ATD/I case.

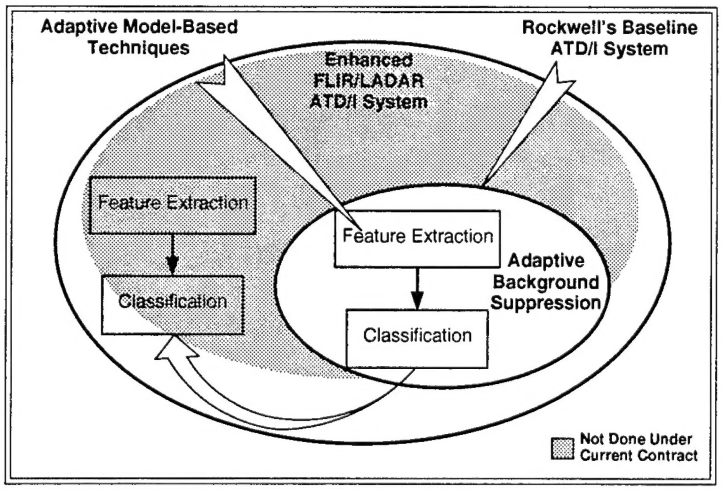

Figure 3. The adaptive background suppression module became the framework with which the proposed enhanced FLIR/LADAR ATD/I system was to have been built. The feature extraction/classification paradigm approach was used. The background suppression module implementation of the approach would be general enough to use for the overall ATD/I system (demonstrated by the arrow pointing upwards to a version outside the background suppression module).

Section 3 is an explanation of the methodology used for this contract and how the feature extraction/classification algorithms affected the overall work. Specifically, this work dealt with log-polar and 3D Hough transform-based feature extraction and hierarchical classification; it is described in Sections 3.1 and 3.2, respectively.

3.1 Feature Extraction

Feature extraction converts the unwieldy output from the sensed image to a manageable size for further processing. Quite often it removes translational, rotational, scaling, and other distortional effects in 2D or 3D. For this application, the basic idea was to map the pre-processed input data into transform space, either log-polar or 3D Hough, before classification.

Sections 3.1.1 and 3.1.2 depict the transform-based approach as it was developed and implemented in FLIR (log-polar transform) and LADAR (3D Hough transform) versions of the background suppression algorithm. In Section 3.1.3, an overall feature vector is proposed that was planned for the enhanced, FLIR/LADAR ATD/I system.

3.1.1 2D FLIR-Based Feature Extraction in Background Suppression

A FLIR-only background suppression algorithm has been developed under this contract that increases overall (FLIR-based) target detection/identification performance (see Section 4.1.1). The rationale for such an algorithm is as follows.

> *It is known that one must lower the target detection threshold to detect faint or hard-to-see targets in FLIR imagery. As a consequence of lowering the threshold, a higher number of clutter objects are introduced to the overall ATD/I system. One is relegated to live with a higher false-alarm rate for FLIR-based target identification in many applications. The background suppression approach developed here will be able to prune false clutter off target candidate lists effectively by identifying objects that are obviously clutter while accepting only targets and few very near target-like clutter objects. The algorithm begins by detecting digital blobs (i.e., indicative of wheeled vehicles, tank turrets, etc.) from FLIR imagery by using a "spoke filter" (see [CGR91] for the foundational paper on the spoke filter and Section 3.1.1.1). It then merges a group of detection hits that are close together (e.g., multiple hits may occur near a target's wheel or track area, etc.). The FLIR-based, background suppression algorithm generates a condensed feature vector based on the shape and the internal gray level to background clutter standard deviation ratio for every candidate object selected by the spoke filter. The technique uses a log-polar transformation process for scale, translation, and rotation invariance. The log-polar output data is then reduced into a manageable feature vector size by selecting bins near a predetermined set of 2D Gaussian-shaped centers called localized receptors. Not only is the output vector invariant to size, translation, and rotation; but, it is also intolerant to small deformations that are due to rotations in depth.*

The next three subsections explain key functions that deal with preprocessing and feature extraction for FLIR-based background suppression. Section 3.1.1.1 describes the role and significance on using a robust blob detector such as the spoke filter. The spoke filter algorithm was taken from previous Rockwell work and tuned for this application.

Sections 3.1.1.2 and 3.1.1.3 define log-polar and coarse coding techniques, respectively. An important example showing how the two functions can produce invariant feature vectors in 2D is shown in Figure 11 of Section 3.1.1.3.

The top level flow diagram of the preprocessing and feature extraction for the background suppression algorithm is shown in Figure 4. Note that the feature extraction box converts likely target object areas into a set of invariant (50-point) feature vectors.

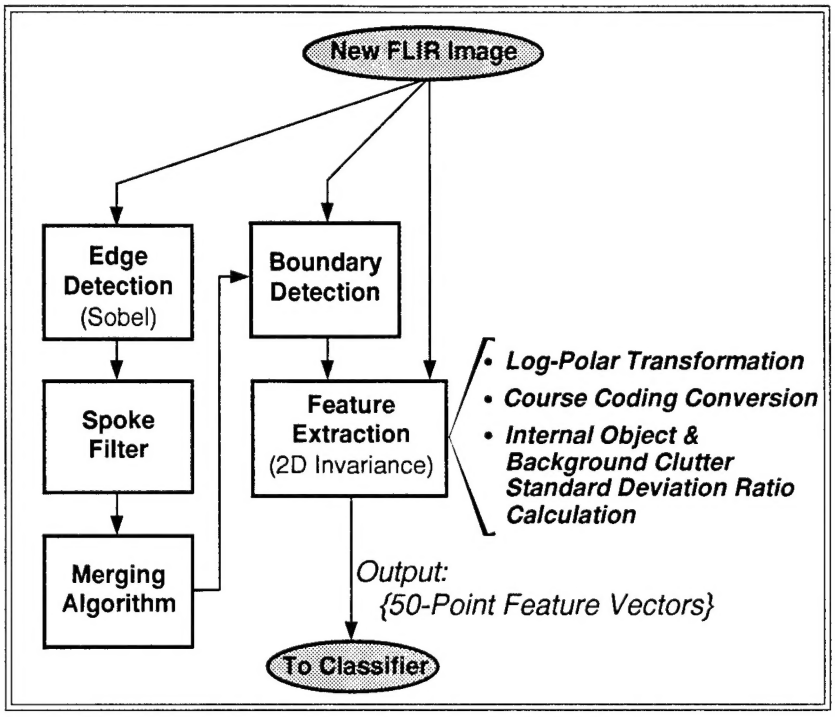

Figure 4. The FLIR-based, background suppression algorithm rejects unwanted clutter objects. The top level flow diagram shown here represents the preprocessing and feature extraction portion of the algorithm.

3.1.1.1 Spoke Filtering Techniques

The spoke filter (developed by Minor and Sklansky in [MS81]) is an extension of the Hough transform for ellipses. The approach assumes that the targets are "blob-like" in nature with distinguishable boundaries not varying wildly from a convex-shaped silhouette. With this assumption, most of the edge segments composing the object-to-background boundary are directed towards the object's geometric centroid. It is a robust and fast approach that researchers are now using in such diverse areas as intelligent vehicle technology, where lane markings are detected in real-time (see Haga *et al.* in [HSK95]).

In this application, the spoke filter converts a narrow or wide Field-Of-View (FOV) FLIR image into a group of silhouettes of candidate targets (or a set of coordinates describing a box that fits around the output silhouette). The algorithm quantizes angle information

from a Sobel edge detector into eight directions as it searches for edges at each angle (the action is analogous to going around a hub of a wheel and examining the spokes).[1]

Figure 5 demonstrates, by example, how spoke filtering is done. Figure 5(a) shows an ideal target to background situation (white circle on a black background). In Figure 5(b), eight rays emanate from the center point of the circle. The rays correspond to the eight quantized angles starting from the horizontal going clockwise. Angles are mapped to the Freeman chain code in the order listed in Table 1. Quantized angles from the ideal target in Figure 5(a) are the smaller, clockwise arrows in 5(b) (as a side note, Figure 5(c) shows the case for the negative image of (a) with small arrows going in a counterclockwise direction). The spoke filter sums the number of different angles produced by the Sobel output for each "spoke". The algorithm stores the sum in an eight-bit "register". In this example, all eight spokes have a representation of three arrows (length of the spoke is equal to three). The output would be a 2D histogram, comprised of eight-bit registers, where the center bin contains the maximum value of eight. The center bin corresponds to the filter's detection of the ideal target. The key point to remember is that one is checking for consistency with a mask similar to Figure 5(b) or (c) on the quantized angle image. Consistency in this context is defined as finding spokes with edge elements (small arrows) aligning themselves in the manner of Figure 5(b) or (c).

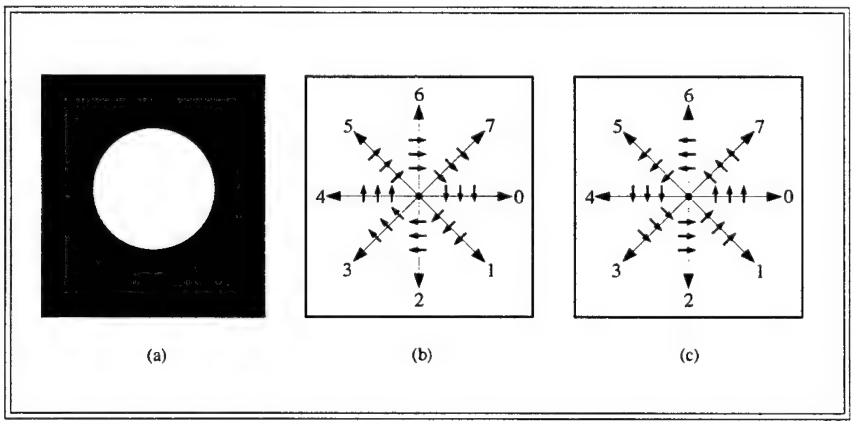

Figure 5. Spoke Filter Directions: (a) positive target, (b) edge output for target brighter than background, and (c) edge output for target darker than background.

When edge elements are aligned, the algorithm sets the *i*th bit, corresponding to its respective quantized angle and chain code. Going back to the example of Figure 5, Figure 6 displays the three register values for the upper right quadrant (spokes 6, 7, and 0). Each group of small arrows from the center point will be counted as setting the appropriate bit to one (where the user designates the length and distance; see the following paragraph). In the example, the 90° spoke will set bit 6 to one, followed by the 45° spoke setting bit 7 to one, *etc.*

[1] It should be noted that 90° is added to every element of the Sobel output. Thus, the vector image (Sobel) output is rotated by 90° in a counterclockwise direction. Since the gradient points in the direction of steepest descent, this operation makes each element point in the direction where the intensity to the *right* of the element is greater than its left. The net effect from all this, is to make the edge elements associated with the blob to align themselves tangentially with respect to its boundary.

Table 1. The spoke filter uses Freeman chain coding in designating its quantized angles.

Quantized Angle (Degrees)	Chain Code Numbers
0	0
-45	1
-90	2
-135	3
180	4
135	5
90	6
45	7

The length of the edge elements and distance from the point in question determine the approximate target size the algorithm will detect. Length, L, and distance, S, from a point (x, y) are measured in pixel widths. In Figure 6, spokes 6 and 0 have $L = 3$ and $S = 1$. The intermediate directions (spokes 1, 3, 5, and 7) are measured in pixel-diagonals (see Minor and Sklansky in [MS81]). The directions are related to L and S by $\acute{L} = \left\lfloor \frac{L}{\sqrt{2}} + \frac{1}{2} \right\rfloor$ and $\acute{S} = \left\lfloor \frac{S}{\sqrt{2}} + \frac{1}{2} \right\rfloor$; where, $\lfloor a \rfloor$ is the largest integer that is not greater than a. Therefore, for spoke 7, $\acute{L} = 3$ and $\acute{S} = 0$.

Other spoke filter details on how it takes care of intersecting blob segments and aggregates blob centroids for improved detection are explained in Minor and Sklansky's paper (in [MS81]). Figure 7 is an example depicting the spoke filter's masking capability. Here it detects circular and elliptical blobs while ignoring line-like clutter.

Once detected areas are found by the spoke filter, additional software is required to (1) label and merge blobs, (2) perform region growing on suspected image areas, and (3) find object boundaries (these functions are included in the merging and boundary detection boxes of Figure 4). The FLIR-based, background suppression algorithm contains vanilla

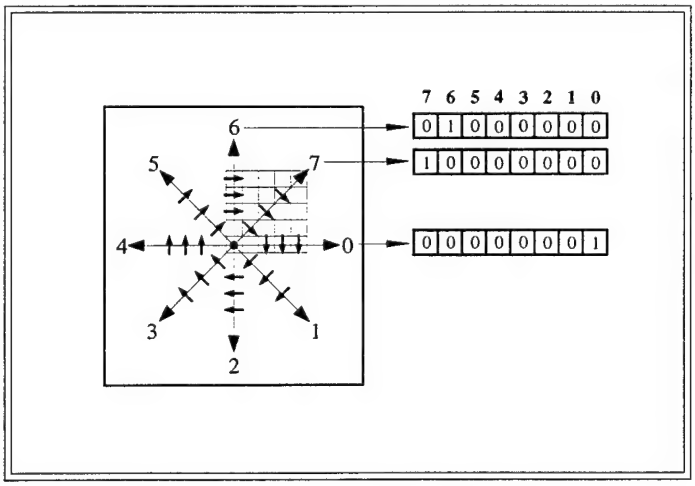

Figure 6. Test example of Figure 5 showing bit setting for spokes 6, 7, 0.

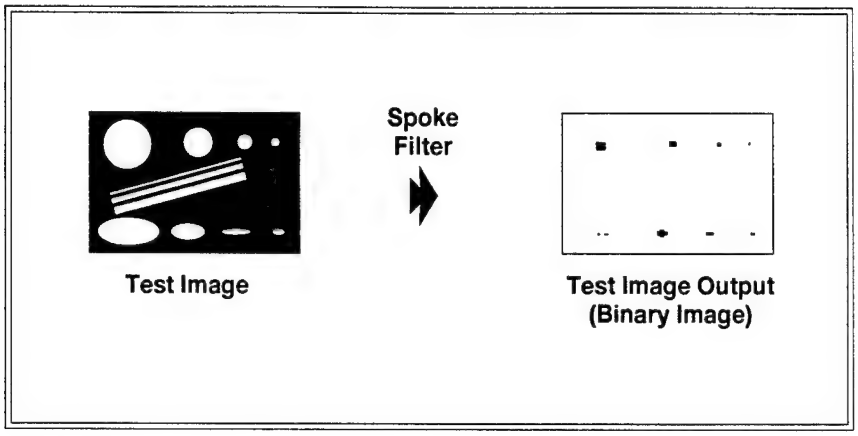

Figure 7. The test image is composed of circular and elliptical blobs with straight lines of varying widths. The spoke filter detects only the corresponding target-like objects.

algorithms to perform merging, region growing, and boundary detecting tasks. One can spend a lot of time implementing exotic algorithms that can improve the capability of these functions, but the emphasis for this contract was placed in implementing innovative feature extraction and classification methods.

Figure 8 shows an example case, at different preprocessng stages of the top level flow diagram in Figure 4, for a FLIR image from the Fort Carson database (see reference [BHPHY94]). Figure 8(a) is a contrast enhanced FLIR image of an M113-109, Armored Personnel Carrier (APC). The vehicle is in the front-end position at approximately 110 meters distance. In Figure 8(b), the spoke filter output is displayed for a radius of $L = 10$ and $S = 1$. Only 5% of the gradient elements from the Sobel operator having moduli greater or equal to the noise threshold T are used in the detection process. The spoke filter detects over 20 likely areas; shown as white blobs in Figure 8(b). Figure 8(c) gives the blob outputs (after merging and segmentation) that contain the boundaries. The segmented output blobs show irregular shapes for most of the clutter objects (which is typical). In Figure 8(d), boundary boxes from the detected objects in Figure8(c) are superimposed on the contrast enhanced original.

In summary, by embedding the preprocessing software with a spoke filter, the FLIR-based, background suppression algorithm can take further advantage of the natural segmentation properties in FLIR imagery.

3.1.1.2 Log-Polar Transformation

After image preprocessing and spoke filtering on the object boundaries, the background suppression algorithm now transforms them from binary subregions into log-polar points. The goal, as explained in the next section, is a feature vector that is composed of a compact or "coarse coded" representation of the log-polar output and the background to object (intensity) standard deviation. Figure 9 pictorially describes the process for a simple tank boundary.

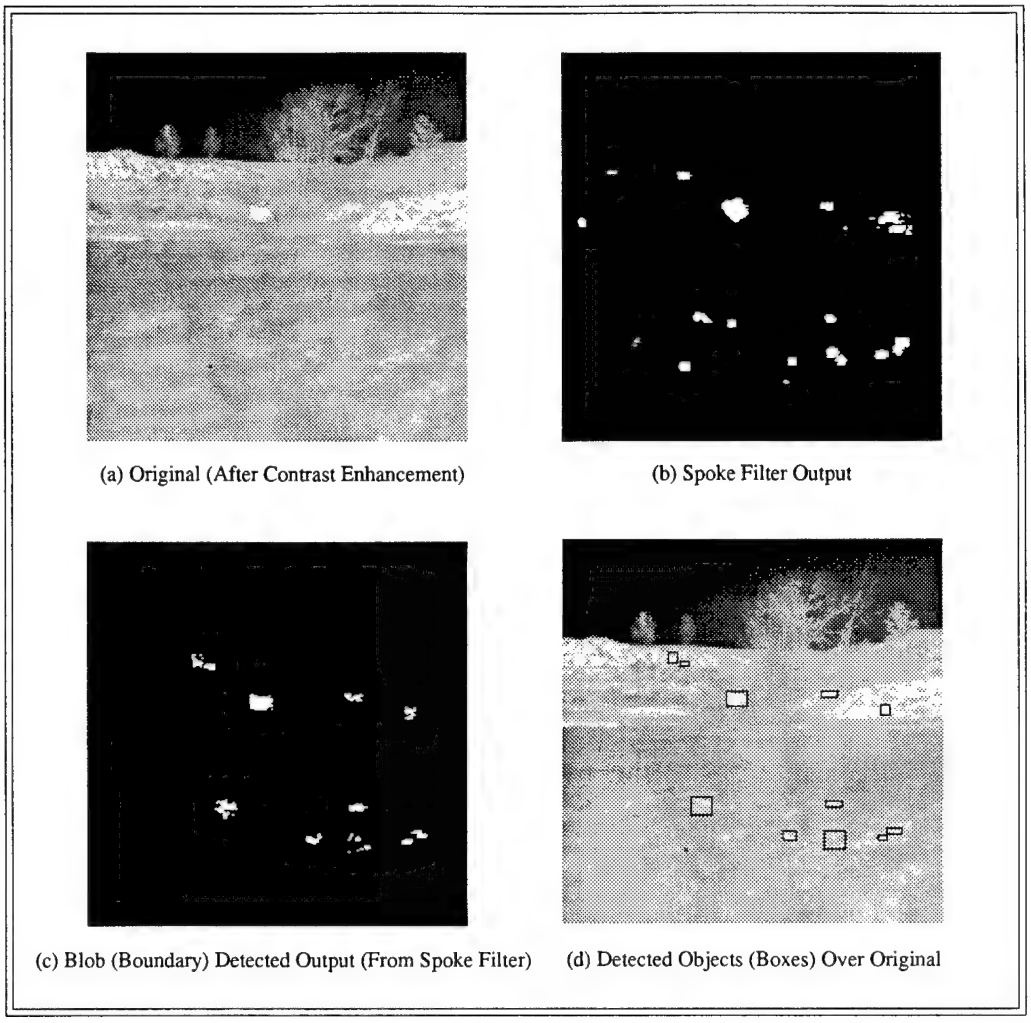

Figure 8. Figure 8(a) shows the contrast enhanced FLIR image (note that the algorithm operated on the original) from the Fort Carson database (see reference [BHPHY94]). The image shows an APC (M113–901) in a front-end position at approximately 110 meters away. Figure 8(b) depicts the binary output (over 20 "white" blobs) from the spoke filter. In Figure 8(c), another binary image displaying the resultant blobs (containing the boundaries) after merging and segmentation. Finally, Figure 8(d) overlays the object boxes found in (c) on to the contrast enhanced original.

In the literature (see the work by Waxman's group at MIT-Lincoln Laboratory in [WSBF93], [WS92], and [BW91])[2], one can transform image points to complex polar space by the following.

For image point (x_i, y_i), a (ρ, θ) space representation is computed by finding the cluster centroid of a set of boundary points, (x_c, y_c). Therefore, its form in (ρ, θ) space is $Z = \rho e^{i\theta}$; where, ρ is defined as the distance of the image point from the cluster centroid,

$$\rho = \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2}\ ;$$

[2] In these references, only high valued curvature points are selected for feature extraction. Here, for more sensitivity, all the boundary points are used in the process.

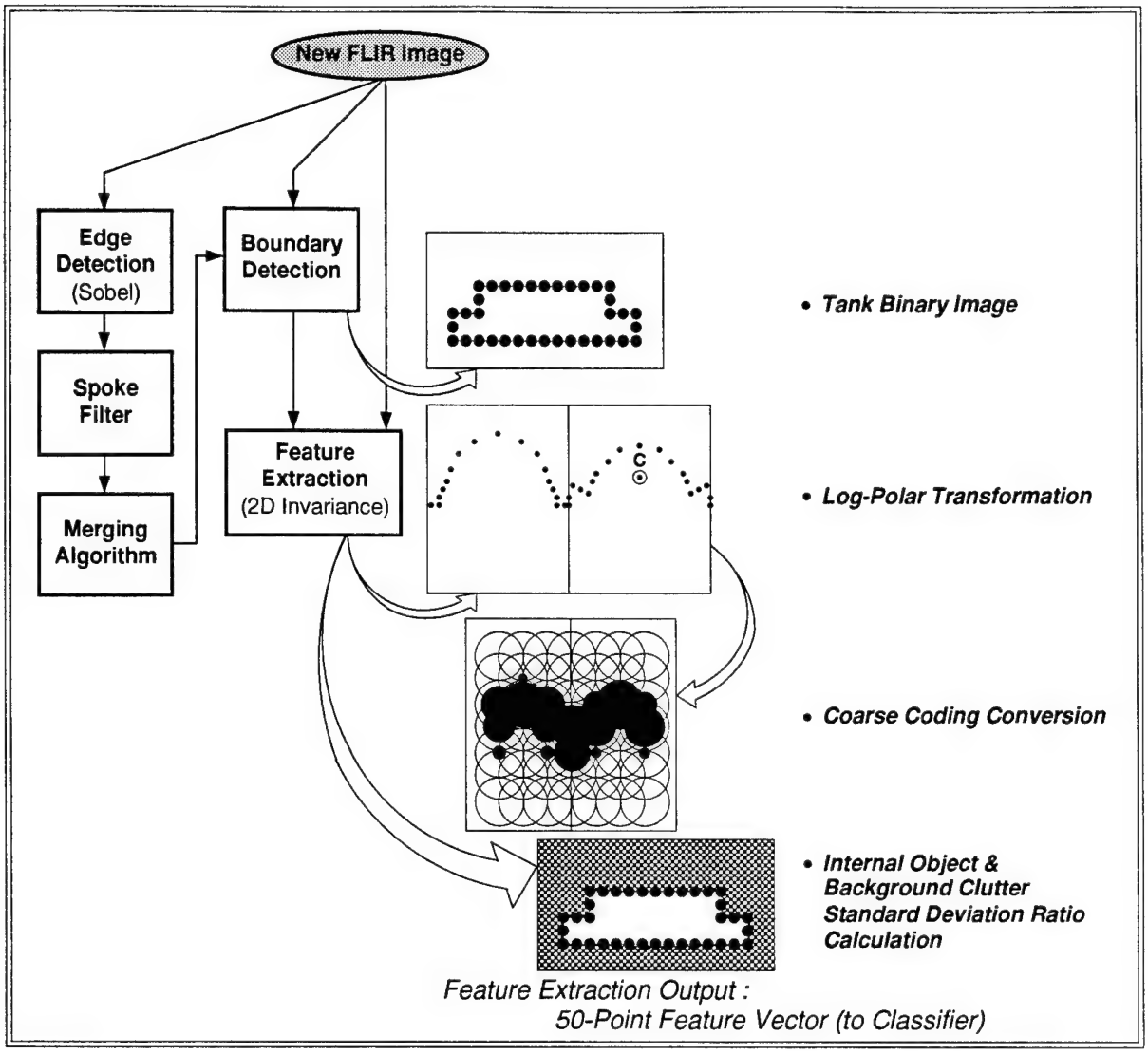

Figure 9. As shown in the above diagram for the simple tank boundary, log-polar transformation combined with coarse coding can produce an invariant feature vector in two dimensions (see Figure 11 for a more detailed description on the process).

and, θ is its angle,

$$\theta = \tan^{-1}\left(\frac{(y_i - y_c)}{(x_i - x_c)}\right).$$

Next, a mapping,

$$ln(Z) = ln(\rho) + i\theta,$$

is applied to the representation. This conformal operation transforms both scaling and rotational changes into **translations** in $(ln(p), \theta)$ space. For example, if the boundary points are rotated by angle θ_r with respect to its group centroid, then $ln(Z) = ln(\rho) + i(\theta + \theta_r)$. In like manner, for a scale factor m, then $ln(mZ) = (ln(\rho) + ln(m)) + i\theta$. After transforming boundary points, one

computes a second cluster centroid of the transformed points. But there is a problem with the second centroid. A simple averaging in the θ-direction will not produce the required centroid. The difficulty is with the 2π periodicity. One solution mentioned by Waxman is to map the feature points onto a complex unit circle $(\cos\theta, i\sin\theta)$; where θ is the same as before. Now the centroid of these points is computed (which will lie outside the complex unit circle): $(\bar{C}, \bar{S})$ such that

$$\bar{C} = \frac{1}{n}\sum_{i=1}^{n}\cos\theta_i$$

and

$$\bar{S} = \frac{1}{n}\sum_{i=1}^{n}\sin\theta_i\,.$$

An effective second centroid, therefore, would be to shift feature points by

$$\tan^{-1}\left(\frac{\bar{S}}{\bar{C}}\right).$$

This operation is performed on the rotated tank boundary for the "log-polar output" row in the illustrative example of Figure 11.[3] The final centroid operation makes the log-polar process for boundaries invariant to scaling, rotational, and translational effects in two dimensional image space.

The next step is to compress the pattern in log-polar space into a finite number of feature vector elements as discussed in the next section. The approach follows the method developed by Waxman *et al.* in references [WSBF93], [WS92], and [BW91].

3.1.1.3 Coarse Coding Techniques in Two Dimensions

Coarse coding techniques are used in the neural net community to provide a compact but effective representation for multidimensional data (see Rumelhart and McClelland in [RM86] for a detailed explanation).[4] In this application, overlaying receptive fields, similar to those shown in Figure 10, are applied to the transformed boundary images. Each field is activated **inversely** to the Gaussian-weighted distance from its center to the closest point. Because the fields overlap, the algorithm has the capability to tolerate small deviations. The output becomes a condensed version of the transformed boundary image. It is invariant to image size, translation, rotation, and some deformation (due to receptors overlapping).

Figure 11 demonstrates the procedure for the simple tank mentioned in Figure 9. In the first row, the tank boundary is expanded to twice its size, rotated by 45° in a counterclockwise direction, and slightly deformed as shown. The second row displays the log-polar operation as described in Section 3.1.1.2. Row three gives the output after the centroid is shifted to the center of the log-polar plot. Note that the second centroid shift,

[3] But it should be noted that for symmetric boundaries (*e.g.*, a square or hexagon), no unique centroid is possible.

[4] Provided, according to Hinton on page 93 in [RM86], the data is sparse. The approach here assumes that taking full boundaries for higher sensitivity rather than selected high curvature points is adequate.

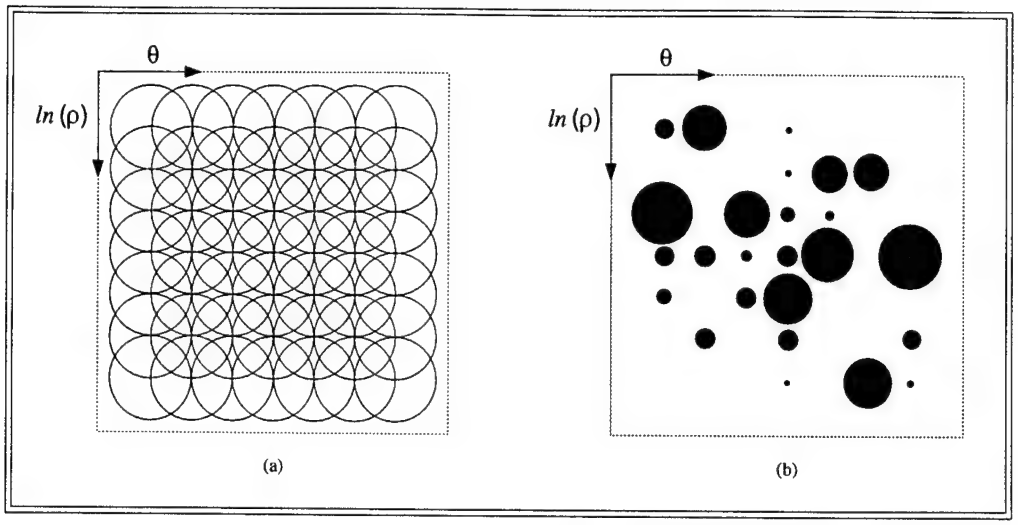

Figure 10. Circular receptive fields are applied to the transformed binary image. Figure 10(a) gives the layout of the receptive fields (for the FLIR-based, background suppression algorithm, it's (7×7)), while Figure 10(b) shows how they operate for a given pattern.

$\tan^{-1}\left(\frac{\bar{S}}{\bar{C}}\right)$, for symmetric boundaries with respect to the y-axis gives 90°. Therefore in practice, one can shift with respect to the 90° point as shown in rows two and three. Coarse coded receptors in the fourth row of Figure 11 indicate the strength of the closest boundary point to its center; where a value close to one represents a filled-in circle while responses near zero are shown by smaller diameters (zero responses are pictured as circle outlines). There is a slight point variation from the ideal in the log-polar plots. The discrepancy is due to the quantization process in the computer simulation. The deformed boundary example (column 4 in Figure 11) can be classified as a target or clutter since it contains some, but not all, attributes of the original. Depending on the application, the deformed boundary may even be indicative of certain types of clutter.

Let

$$\mathbf{F} = \begin{bmatrix} f_0 \\ \vdots \\ f_{49} \end{bmatrix}$$

be the 50-element background suppression feature vector. The first 49 components are reserved for the overlapping receptor outputs. Define f_0 to be the value for top left receptor output. Feature elements, $f_1, \ldots, f_{48}$, correspond to receptor fields going from left to right and top to bottom.

In order to take advantage of the obvious segmentation capability between hot/cold objects and their background in FLIR imagery, the 50^{th} feature element, or f_{49}, contains a measure of this sensitivity. That is,

$$f_{49} = \frac{\sigma_o}{\sigma_b};$$

where, σ_o is the object (intensity) standard deviation and σ_b is the background (intensity) standard deviation from a (16×224) pixel strip near a border of the (256×256) FLIR image.

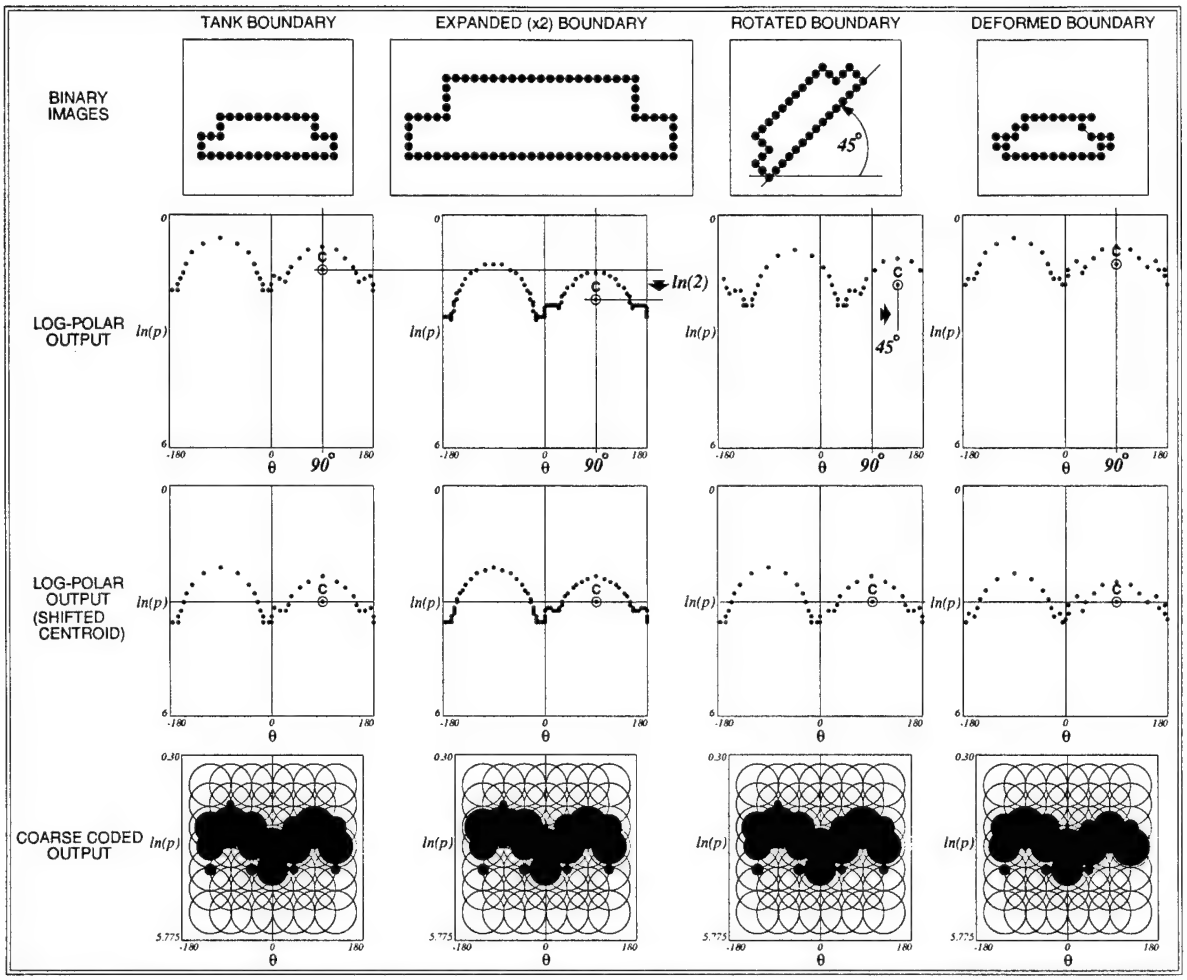

Figure 11. For the simple tank example (top left), columns 2 and 3 show 2D invariance in translation and rotation (point variation is due to quantization in the binary images and in the log-polar plotting process). Small amounts of deformation in the fourth column produce a similar coarse coded output to the original tank boundary. Depending on the model-based library and application, the back-end classifier may designate the deformed boundary example as a representative from the original class, another target model, or clutter. Also, note the arrows for the expanded and rotated boundaries (columns 2 and 3 of row 2). They show their corresponding movement from the second centroid (*i.e.*, $\tan^{-1}\left(\frac{\bar{S}}{\bar{C}}\right)$; see Section 3.1.1.2) of the original tank boundary.

To continue with the FLIR-based, background suppression algorithm, the reader should go to Sections 3.2 and 3.2.1 for a discussion on the back-end classifier. For performance results, see Section 4.1. In the next section, a 3D LADAR-based, feature extraction approach is presented.

3.1.2 3D LADAR-Based, Feature Extraction in Background Suppression

During the first half of 1994, work began on a 3D LADAR-based, background suppression algorithm that culminated with a lab demonstration on a SUN workstation for Demo B (at Lockheed Martin, Denver on June 28 and 30, 1994). The lab demo showed a new method to suppress background clutter by implementing the feature extraction/classification

paradigm mentioned in the beginning of Section 3. In the work, LADAR range imagery[5] was used to classify targets from non-targets (or clutter) with good results (see LADAR-based background suppression results in Section 4.1.2). The idea demonstrated was that clutter suppression can be accomplished by identifying target or clutter objects with a trained classifier. Such a system incorporated the observation that natural clutter structures have, on average, smaller area planes with their normals scattered over a wider orientation range than man-made objects.[6]

The LADAR-based, background suppression algorithm compared sensed data to the predicted target signature and background clutter. This task is implemented by converting the range imagery of unknown objects to 3D plane primitives for transformation into a 3D histogram. The histogram for an unknown object was then converted to a feature vector through coarse coding techniques. The process is similar to what was described in the 2D log-polar and coarse coding sections (Sections 3.1.1.2 and 3.1.1.3, respectively). The technique for 3D histogram generation is based on work by Krishnapuram and Casasent (in reference [KC89]) for their 3D Hough transform. What is done here is to generate 3D Hough transforms for unknown objects. In an analogous manner to Sections 3.1.1.2 and 3.1.1.3, one uses 3D Hough space to generate feature vectors. The process by which feature vectors are produced is by overlaying receptors in 3D over Hough transform space (which is shown pictorially in Figure 12).

To be more specific (and in following Krishnapuram and Casasent's development in reference [KC89]), assume a set of unit vectors of the form

$$\mathbf{u}_n = a_n\mathbf{i} + b_n\mathbf{j} + c_n\mathbf{k}$$

in 3D image space; where,

$$\sqrt{a_n^2 + b_n^2 + c_n^2} = 1 \, .$$

Given the above definition, one can define the 3D Hough transform for a typical vector $\mathbf{r}$ as

$$p = \mathbf{r} \cdot \mathbf{u}_n \, .$$

Note that n and p are the parameters that describe direction in 3D and distance, respectively. In order to practically implement the approach, the number of values for p and unit vectors for $\mathbf{u}_n$ are limited to a finite amount. Now for plane detection, let the LADAR range image be defined as

$$I(x, y) = z \, ;$$

[5] The imagery consisted of seven objects, 35 target and 35 clutter, that were selected manually from the LADAR image set of the Fort Carson database (see reference [BHPHY94]).

[6] There are many instances where it is easier to separate natural and man-made objects. For example, in this domain, military vehicles tend to have higher valued bins grouped together; while certain natural objects such as trees have smaller valued bins randomly scattered throughout the histogram. The random orientation of planes in the trees is due to the LADAR sensor picking up gaps between branches thereby limiting the build-up of large planar surfaces that are characteristic of many man-made objects.

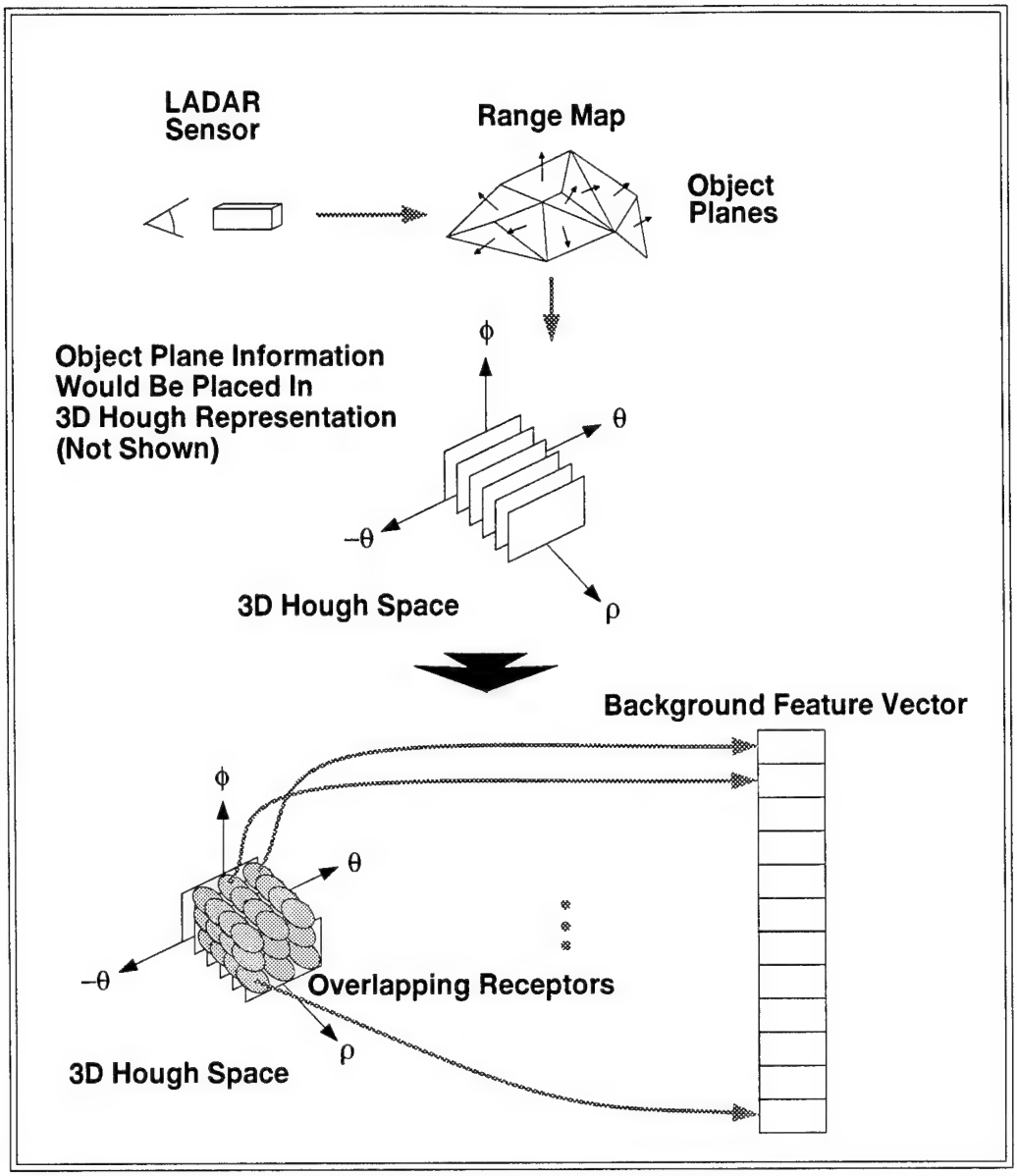

Figure 12. The LADAR-based, feature extraction process takes advantage of range map information in order to build a representative feature vector. In this figure, an object in a LADAR image is converted to a list of object planes of a normal unit vector and 3D location on its way to a 3D Hough transform representation (not shown). As explained in Section 3.1.2, an object's 3D Hough representation is the basis for an analogous coarse coding technique that was done in 2D to arrive at a background feature vector.

where, z is the intensity for each (x, y) pixel. Assume a plane P in 3D space. For each pixel point (x, y) in the image, one can construct a vector,

$$\mathbf{r} = x\mathbf{i} + y\mathbf{j} + z\mathbf{k}\,.$$

All vectors $\mathbf{r}$ (that are points on the plane P), with a unit vector $\mathbf{u}_n$ perpendicular to P, produce the same values for $p = \mathbf{r} \cdot \mathbf{u}_n$. Therefore, all points on plane P will vote for the same point (n, p) in 3D Hough transform space. This action will generate a peak in the space. The conclusion that is reached by constructing such a space for an object is that

if one locates the peaks in 3D Hough space, then a description of the orientations and sizes of the planes for that object are found.[7] Krishnapuram and Casasent (in reference [KC89]) go on from this point to give details on how one should go about implementing the 3D Hough transform in practice for 3D object location and recognition.

But in this work, the intent was to obtain a feature vector that is representative of its underlying plane structure. As in the 2D case (of Section 3.1.1.3), overlaying receptive fields are applied to the 3D Hough space for characterization. Each field is activated inversely to the Gaussian-weighted distance from its center to the closest point. Figure 12 shows how the fields overlay in 3D Hough space. Therefore, the object's representation in 3D Hough space (not shown in Figure 12) would be in the form of peaks due to the number of planes voting for that particular bin. The peaks that are the closest to a receptor would stimulate it in an inverse way to the distance from its center. The net effect would be a feature vector that is a characterization of an object's plane information. The top level process is summarized and shown in Figure 12. The experience gained with this technique shows that internal target structure adds to robustness of the overall background suppression algorithm when compared to using only target silhouette (or boundary) information.

To continue with the LADAR-based, background suppression algorithm, the reader should go to Section 3.2.2 for the back-end classification algorithms and Section 4.1.2 for the results obtained with the overall algorithm suite. In the next section, the proposed, FLIR/LADAR ATD/I is introduced.

3.1.3 Proposed Feature Extraction in Enhanced, FLIR/LADAR ATD/I System

The approach taken for this contract was to develop a state of the art ATD/I system by building on the foundation of the feature extraction/classification paradigm. This system would have employed the 2D and 3D feature extraction techniques described in Sections 3.1.1 and 3.1.2, respectively. But as discussed in Section 2.1, this objective was not achieved due to program funding restrictions. Despite the setback, careful thought was given during the coarse of the contract on the nature of such a target identification system. For the overall system, two key ideas emerged: (1) to embed the system with a hierarchical classification algorithm that is similar to what is described in Section 3.2 and (2) to blend new and unexpected visual experiences (from targets) on the fly after sufficient training with a CAD model set. The enhanced ATD/I system would be capable in deciding between targets that have "distinctive features" (*e.g.*, such as a longer gun barrel). It would decide through evidence accumulation which target class best fits the new image features that are encountered. The network would not only identify unknown targets from "snapshots" (sustained views from targets), but also from aspect sequences in the input FLIR/LADAR data stream. Both static and dynamic target identification situations would be enhanced because neighboring aspect views assist in the total recognition process. These ideas are summarized and partitioned into two segments: preprocessing/feature extraction (this section) and target identification (in Section 3.2.3).

[7] The object description would have n giving the plane orientation, p its perpendicular distance from the origin, and the height of the peak being proportional to the number of points in each plane.

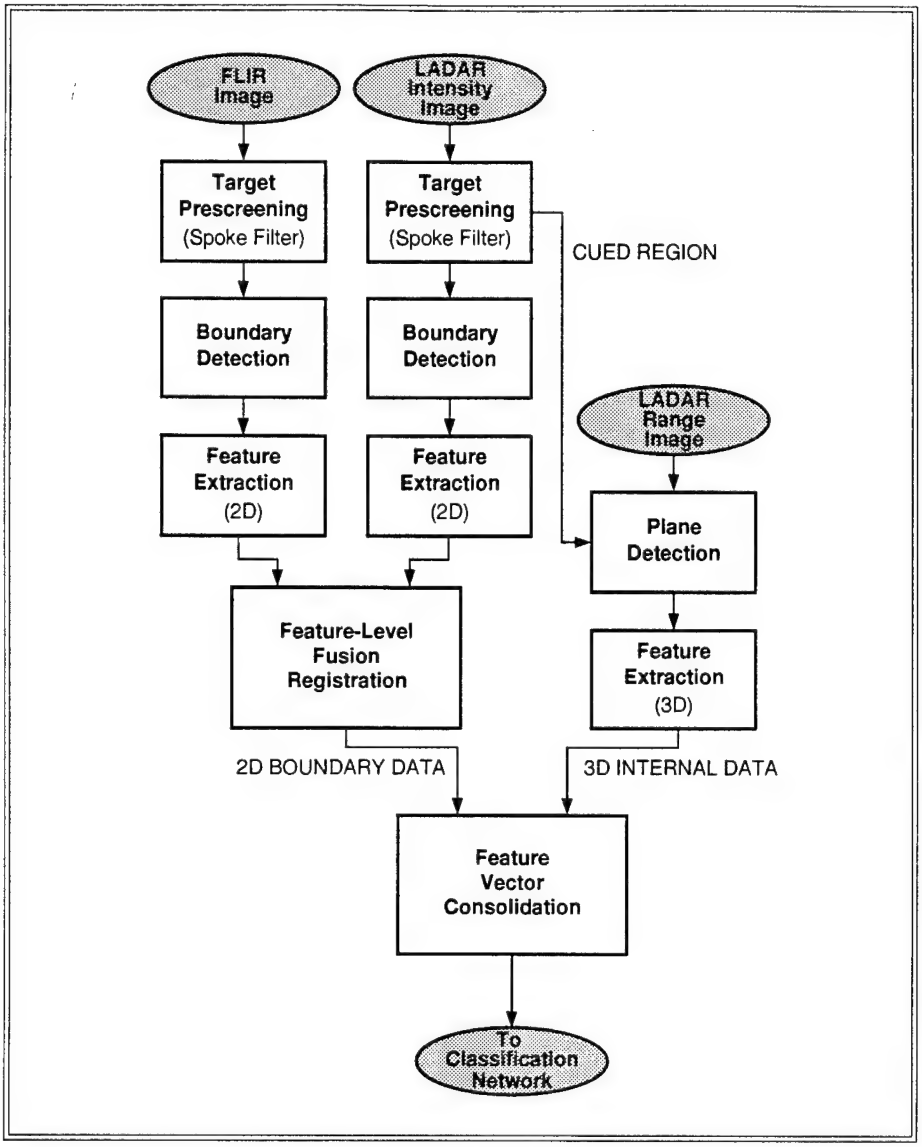

Figure 13. The proposed preprocessing and feature extraction approach for the enhanced, FLIR/LADAR ATD/I system fuses FLIR and LADAR data on a feature level. See Figure 23 for the overall system diagram.

Figure 13 shows a flow diagram of the preprocessing and feature extraction portion of the proposed FLIR/LADAR ATD/I system. In the diagram, FLIR and LADAR intensity imagery are fed into the target prescreening modules. These modules will be spoke filter units. The idea here is to take advantage of the blob-like nature of many tactical targets. After detecting the object boundary, feature extraction processing in two dimensions is performed. The techniques used would be similar to what was described in Sections 3.1.1.2 and 3.1.1.3. Fusing the two boundaries is envisioned in the feature-level registration box of Figure 13. Matching can be done by using minimum mean square error or hierarchical classification techniques.[8] After matching, the system

[8] In the hierarchical classification case, the system would employ the general method of the learning system introduced in Section 3.2; *i.e.*, searching for the best match by comparing boundaries down a tree structure in feature space.

performs a blending operation of the feature elements; where, the percentages may be a function of the application and environment. **Because the LADAR intensity image and range map are by definition co-registered, the corresponding 3D range image can be processed with the Hough-based, feature extraction techniques of Section 3.1.2. One can then say both boundary and internal data are processed by this method.** The boundary information will be in the form of a n-element feature vector from the fused FLIR/LADAR intensity imagery. The internal data will be a m-element vector stemming from the plane information found in the LADAR range map. The feature vector consolidation box, in Figure 13, would just append the two inputs.

This new ATD/I approach would have the capability to robustly fuse two different complimentary inputs. Registration on a feature level would be accomplished quickly and effectively.

For further description on the enhanced system, see the target identification summary in Section 3.2.3.

3.2 Classification: The Hierarchical Approach

Classification theory is the study of the ways to categorize data. It is an attempt to mimic what human beings do: generalize and abstract from specific examples, discriminate similar patterns by some measure of performance, and store/recall information.

Hierarchical classification, in particular, incorporates a graded structure to accomplish categorization. There are advantages to framing classification in such a structure. Researchers such as Jose Ambros-Ingerson, Richard Granger, and Gary Lynch (in [AIGL90]) have stated that human subjects in perceptual studies robustly recognize objects first as categorical levels and subsequently at successively subordinate levels. They further state that such studies suggest a presence of structured memories that are organized and searched hierarchically during recognition.

The rationale behind a hierarchical approach can be summarized in the following statement.

> *The brain processes information by using a principle of contrast; that is, suppressing information that does not change and enhancing parts that do. One implementation of this idea is to cast the operation in terms of differences. Construct an algorithm that maneuvers through a tree structure as it finds the correct classification. It will navigate down the tree by subtracting the residual portion of the unknown vector by checking the current cluster centroid (or prototype) vector in deciding what path the algorithm should take.*

Figure 14 describes what is intended in combining hierarchical classification system in a background clutter suppression context.

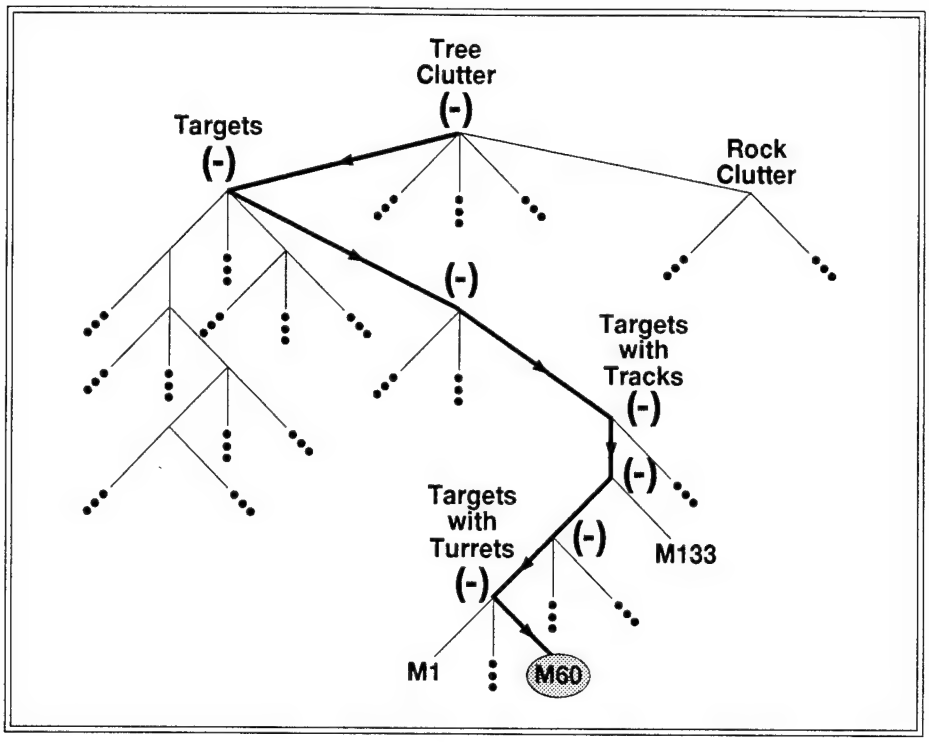

Figure 14. One approach in a classification hierarchical system would be to subtract the current feature vector with cluster prototypes as the algorithm maneuvers through weight space. The goal above is to correctly identify the M60 tank through differencing.

As mentioned in the beginning of Section 3, the feature extraction/classification paradigm was selected to be the foundation upon, which ATD/I work would be done for the contract. In a manner similar to Section 2.1, Section 3.2 will describe how hierarchical classification was applied in this program. Sections 3.2.1 and 3.2.3 describe background suppression and the enhanced ATD/I system algorithmic implementation, respectively. In the background suppression section, emphasis is placed on the 1995 FLIR-based development. The ATD/I system section contains a discussion on how the transform-based feature extraction and hierarchical adaptive differencing would have worked.

Hierarchical classification was performed on both FLIR-based and LADAR-based clutter suppression. The objective was to prune off obvious clutter objects from target candidate lists. These lists would in turn be used as input to target identification algorithms.

3.2.1 Classification in FLIR-Based Background Suppression

The objective with FLIR-based background suppression is to classify target versus clutter objects after spoke-based preprocessing and feature extraction have been performed on the selected subregions of the input image (see Section 3.1.1). To push the state of the art in FLIR target detection/clutter suppression a hierarchical approach was tried and tested. In the FLIR-based version, a hierarchical clustering algorithm (with the kernel based on Carpenter and Grossberg's ART 2-A neural net in [CGR91]) classifies unknown objects into background clutter or targets after training on a representative set of feature vectors. Figure 15 adds the classification portion to the diagram shown in Figure 4.

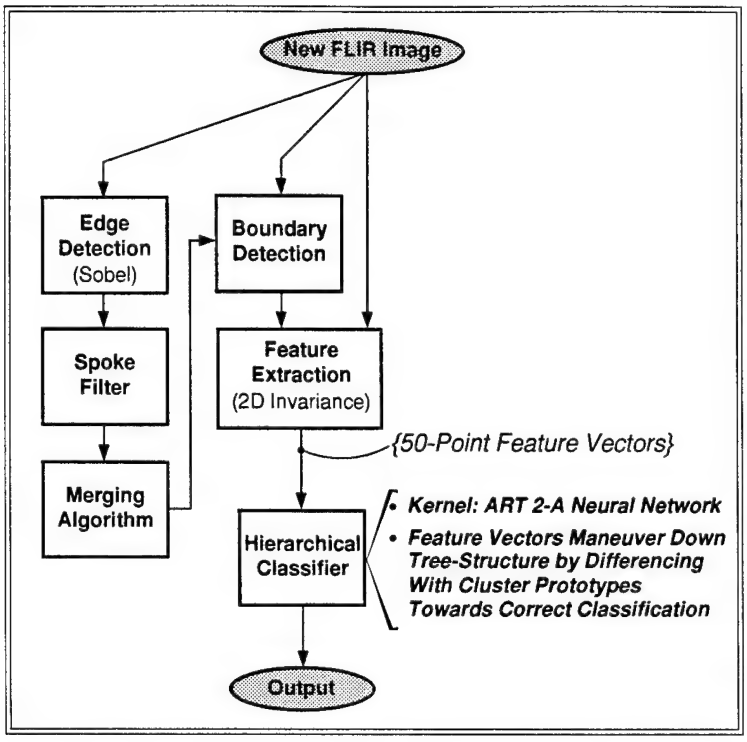

Figure 15. The top level flow diagram of the FLIR-based, background suppression algorithm now is completed with the hierarchical classification module.

3.2.1.1 ART 2-A

The kernel algorithm for the hierarchical classification module is the ART 2-A neural network. The ART 2-A system (see [CGR91]) is a fast algorithmic form of Carpenter/Grossberg's ART 2 neural network that was developed primarily for analog input patterns (see [CG87]). It is an **unsupervised** neural network that generates output clusters that are especially suited for the purpose proposed in this work. It also handles large databases (unlike many other neural network paradigms).

The following paragraphs are an explanation of the step-by-step description of the algorithm in Figure 16. It is a composition of references [CGR91], [TG94], and [FKMHH92].

Embedded Neural Network: ART 2-A. From Figure 16, there are two active layers in the network: F_1 and F_2. The F_1 layer is a preprocessing module that performs noise suppression and contrast enhancement on the feature vectors. Define an M-element input vector, $\mathbf{I}^0$. Let $\mathbf{I}$ be the output from the F_1 layer. The F_1 layer processes the $\mathbf{I}^0$ vector by

$$\mathbf{I} = \eta f_\theta \eta \mathbf{I}^0;$$

where, η normalizes any vector to its unit vector and f_θ removes vector elements below the threshold, θ (see Step 2 of Figure 16).

1. Intialize:

$$0 \leq \rho^* \leq 1,$$
$$\mathcal{L}_j = false,$$

and,

$$0 \leq \beta \leq 1.$$

(Note, for $\beta = 0$, the network always selects the winning weight vector, $\mathbf{z}_J^{*(new)}$; see **Step 7.**)

2. Transform the input pattern from F_0 to F_1 such that, for $\mathbf{I} = (I_0, \ldots, I_{M-1})$,

$$\mathbf{I} = \eta f_\theta \eta \mathbf{I}^0;$$

where,

$$\eta \mathbf{x} = \frac{\mathbf{x}}{\|\mathbf{x}\|},$$

$$(f_\theta \mathbf{x})_i = \begin{cases} x_i & \text{if } x_i > \theta \\ 0 & \text{otherwise} \end{cases}$$

(for $i = 0, \ldots, M-1$), and

$$0 \leq \theta \leq \frac{1}{\sqrt{M}}.$$

3. Activate F_2 by the following:

$$T_j = \begin{cases} \alpha \sum_{i=0}^{M-1} I_i & \text{if } \mathcal{L}_j = false \\ \sum_{i=0}^{M-1} I_i z_{ji}^* & \text{if } \mathcal{L}_j = true \end{cases};$$

where, the constant α must be:

$$\alpha \leq \frac{1}{\sqrt{M}}.$$

4. Choose the best matching category:

$$T_J = \max \{T_j : j = 0, \ldots, N-1\}.$$

(Note, if two or more nodes are the same value, then choose one at random.)

5. Test for vigilance:

if $(\mathcal{L}_J = true$ and $T_J \geq \rho^*)$

OR

if $(\mathcal{L}_J = false)$

THEN *goto* **Step 7.**

6. Reset node J to index of arbitrary uncommitted node (remember that all uncommitted nodes have $\mathcal{L}_J = false$).

7. Adapt weights for winning F_2 node:

$$\mathbf{z}_J^{*(new)} = \begin{cases} \mathbf{I} & \text{if } \mathcal{L}_J = false \\ \eta\left(\beta\eta\Psi + (1-\beta)\mathbf{z}_J^{*(old)}\right) & \text{if } \mathcal{L}_J = true \end{cases};$$

where, for $i = 0, \ldots, M-1$,

$$\Psi_i = \begin{cases} I_i & \text{if } z_{Ji}^{*(old)} > \theta \\ 0 & \text{otherwise} \end{cases}$$

(with $\mathbf{z}_J^{*(old)}$ being the value for $\mathbf{z}_J^{*(new)}$ at the beginning of the input presentation).

Adjust the J^{th} F_2 node status:

if $(\mathcal{L}_J = false)$

THEN $(\mathcal{L}_J = true)$.

8. *Goto* **Step 2.**

Figure 16. The ART 2-A neural network drives the differencing technique employed by the FLIR-based, background suppression algorithm.

In the N-node layer, F_2, the best matching category is obtained by finding the maximum value for the set of activation parameters. If $\mathbf{z}_j^*$, such that

$$\mathbf{z}_j^* = \left(z_{j0}^*, z_{j1}^*, \ldots, z_{j(M-1)}^*\right),$$

represents the weight (or cluster prototype) vector for node j,[9] then T_j is the activation value with T_J being the maximum value as shown in Step 4 (of Figure 16). This result is seen from Step 3, where the F_2 layer computes

$$T_j = \mathbf{I} \cdot \mathbf{z}_j^*$$

for a committed node and

$$T_j = \alpha \sum_{i=0}^{M-1} I_i$$

for an uncommitted one.[10]

If p^* denotes the vigilance of the ART 2-A network,[11] $\mathbf{z}_J^*$ the weight vector for node J, and β the learning rate, then only winning node J will have its weights changed. Therefore, if $T_J \geq \rho^*$, then according to Step 7,

$$\mathbf{z}_J^{*(new)} = \eta\left(\beta\eta\boldsymbol{\Psi} + (1-\beta)\,\mathbf{z}_J^{*(old)}\right);$$

otherwise, the network allocates a new node $\mathbf{I}$ with

$$\mathbf{z}_J^{*(new)} = \mathbf{I}.$$

Here, the vector $\boldsymbol{\Psi}$ is the union of the current input and weight vectors (as shown in Step 7) with $\mathbf{z}_J^{*(old)}$ defined as the value for $\mathbf{z}_J^{*(new)}$ at the beginning of the input presentation. It should be noted, that the system through the learning law automatically generates new F_2 nodes (or cluster prototypes) when necessary.

It is the ART 2-A neural network that drives the FLIR-based, background suppression algorithm into pruning background clutter from the target candidate list. Now the task is to come up with a scheme for embedding the neural network in a differencing structure.

---

[9] Asterisks for the weight vector and the vigilance parameter (defined in the next paragraph) relate to ART 2 terms found in [CGR91]. It is not necessary for understanding the algorithm; *e.g.*, references [TG94], and [FKMHH92] do not have them.

[10] An F_2 node where learning has gone on before is defined as a "committed" node; otherwise, it is considered to be "uncommitted" (with no prior learning taken place).

[11] A vigilance value near one makes the network very selective, while a number close to zero produces a network with little discrimination between classes.

3.2.1.2 Maneuvering Through Feature Space

This section describes the method used to embed the ART 2-A algorithm in a hierarchical control structure for classifying targets and clutter. The intent was to form the foundation for the recognition engine on the enhanced, ATD/I system by developing and testing the hierarchical technique on the FLIR-based, background suppression algorithm. The goal is to prune off obvious clutter objects from the target candidate list. The important new part to the overall classification algorithm implemented here is to incorporate differencing techniques where the residual portions of the input feature vector are subtracted by the current cluster prototype (*i.e.*, $\mathbf{z}_j^*$) chosen by the system. The procedure that follows is motivated by the work of Ambros-Ingerson, Granger, and Lynch at the Center for the Neurobiology of Learning and Memory, University of California - Irvine (see [AIGL90]). In addition, the approach has some similarities with Borsi *et al.* at the University of Hannover; where, the application is to recognize faults in high-voltage systems (in [BGW95]). Borsi cascaded ART 2-A networks for hierarchical clustering, but he didn't employ a differencing scheme.

The hierarchical classification algorithm is explained in two phases: training and test. In a manner similar to the ART 2-A algorithm section (in 3.2.1.1), the training and test parts are depicted with (1) a concise algorithm listing and (2) a brief commentary on the step-by-step description. To elucidate the method even more, an illustrative example is included with the commentary.

Hierarchical Classification: Training Phase. As shown in the listing of Figure 17 and the illustrative example in Figure 18, a tree structure is created for the set of input feature vectors, $\left\{\mathbf{F}_0, \mathbf{F}_1, \ldots, \mathbf{F}_{(N_k-1)}\right\}$; where, N_k is the total number of feature vectors to be trained during this session.[12] The algorithm processes the feature vector database serially as instructed by Step 1 of Figure 17. In Step 2, the algorithm sets Level l to zero and stores the input feature vector into the current residual (feature vector) array, $\mathbf{F}_k^{(l)}$.

After initialization, the program learns the input feature vector via the ART 2-A neural net (see Step 3). Learning may take several passes with the current residual feature vector set that is associated with the particular node in question.[13] From Step 3, the vigilance parameter, ρ^*, is the same for each Level l down the tree; but, it can be made to vary.[14] The training process per node and Level l is symbolically depicted in Figure 18 with the appropriate label and arrows shown on the right side of the diagram.

---

[12] For Demo C and the ICD version delivered to Lockheed Martin, N_k would correspond to the number of detections found per (256×256) FLIR image.

[13] In the training software that generated the tree structures for the Demo C and ICD background suppression programs, three passes from the residual vectors in the same order that established the node was enough for stabilization.

[14] For example, depending on sensitivity to discover certain secondary structure for an input database, ρ^*, can be different for every node and level on the tree. In the training software that generated the tree structure for the ICD version, three different vigilance values were used for "mixed clusters": $\rho^* = 0.85$ for Level 0, $\rho^* = 0.99$ for clusters composed of two residual vectors, and $\rho^* = 0.77$ for all the rest.

Next, from Step 4, the winning cluster prototype, $\mathbf{z}_J^{*(l)}$, for node index J is identified after the ART 2-A training process on Level l. Step 5 is a logic statement that describes the algorithm's action after training. If winning node J is the same class as the feature vector, the algorithm is finished with it (*i.e.*, $\mathbf{F}_k$ has influenced the system by having the winning clusters down its path through the tree adapt to its features). The program then returns to Step 1 for feature vector, $\mathbf{F}_{k+1}$. If there is a misassociation in classes or the node is considered "mixed",[15] then the algorithm computes the residual feature vector for the next level (in Step 6).

Step 6 performs a subtraction and absolute value operation between the residual feature vector, $\mathbf{F}_k^{(l)}$, and the winning cluster prototype, $\mathbf{z}_J^{*(l)}$, on level l such that,

$$\mathbf{F}_k^{(l+1)} = \left| \mathbf{F}_k^{(l)} - \mathbf{z}_J^{*(l)} \right|.$$

The main reason for the absolute value operation is due to an ART 2-A requirement. The neural network must have all of its (input vector) elements be greater or equal to zero. There's no harm by taking the absolute value since the algorithm is still "masking" out characteristics by differencing. Indeed, in reference [AIGL90], a more generalized version of their algorithm replaces the subtraction operation with a "masking" term.

Steps 7 and 8 make it possible for the algorithm to create a tree structure by incrementing l as it goes back to Step 3 for more training through the ART 2-A neural network.

Once all the feature vectors are processed by the training algorithm listed in Figure 17, a tree structure is created that may look similar to what is shown in Figure 18.

Several observations concerning the ART 2-A embedded, hierarchical training process can now be made.

1. **All** training vectors have a final (non-mixed) node that represents their class. In other words, 100% correct classification is achieved during training.
2. Classification generalization of the neural network increases as one decreases the vigilance parameter, ρ^*. This capability is based on the definition and purpose of vigilance in the ART 2 and ART 2-A networks. For a more detailed explanation, see references [CG87] and [CGR91].
3. The algorithm has the potential of identifying hierarchical structure in the training database (see [AIGL90]).

[15] A cluster is considered "mixed" when it represents vectors from both classes.

1. Do **Steps 2 - 8** for each input feature vector, $\mathbf{F}_k$, $k = 0, 1, \ldots, (N_k - 1)$; where, N_k is the total number of feature vectors to be trained.

2. Set Level $l = 0$ and $\mathbf{F}_k^{(l)} = \mathbf{F}_k$.

3. Train vector $\mathbf{F}_k^{(l)}$ with the ART 2-A network preset at a vigilance of ρ^*.

   (Note, to adequately train with the ART 2-A network, it may necessitate running the residual training vector set, from the winning cluster, J, on Level l, through several times until its category structure stabilizes.)

4. Identify winning cluster prototype, $\mathbf{z}_J^{*(l)}$; where, J is the winning node index.

5. If the cluster associated with $\mathbf{z}_J^{*(l)}$ is the same class as $\mathbf{F}_k$ *goto* **Step 1** (the algorithm is finished with the current feature vector); otherwise, if it hasn't already been so designated, label the cluster as "mixed".

   (Obviously, a cluster is considered "mixed" when it represents vectors from both classes.)

6. Compute the residual feature vector,

$$\mathbf{F}_k^{(l+1)} = \left| \mathbf{F}_k^{(l)} - \mathbf{z}_J^{*(l)} \right|.$$

7. Set $l = l + 1$.

8. *Goto* **Step 3**.

Figure 17. The control structure for the FLIR-based, background suppression algorithm is created during the training phase by adapting a differencing paradigm.

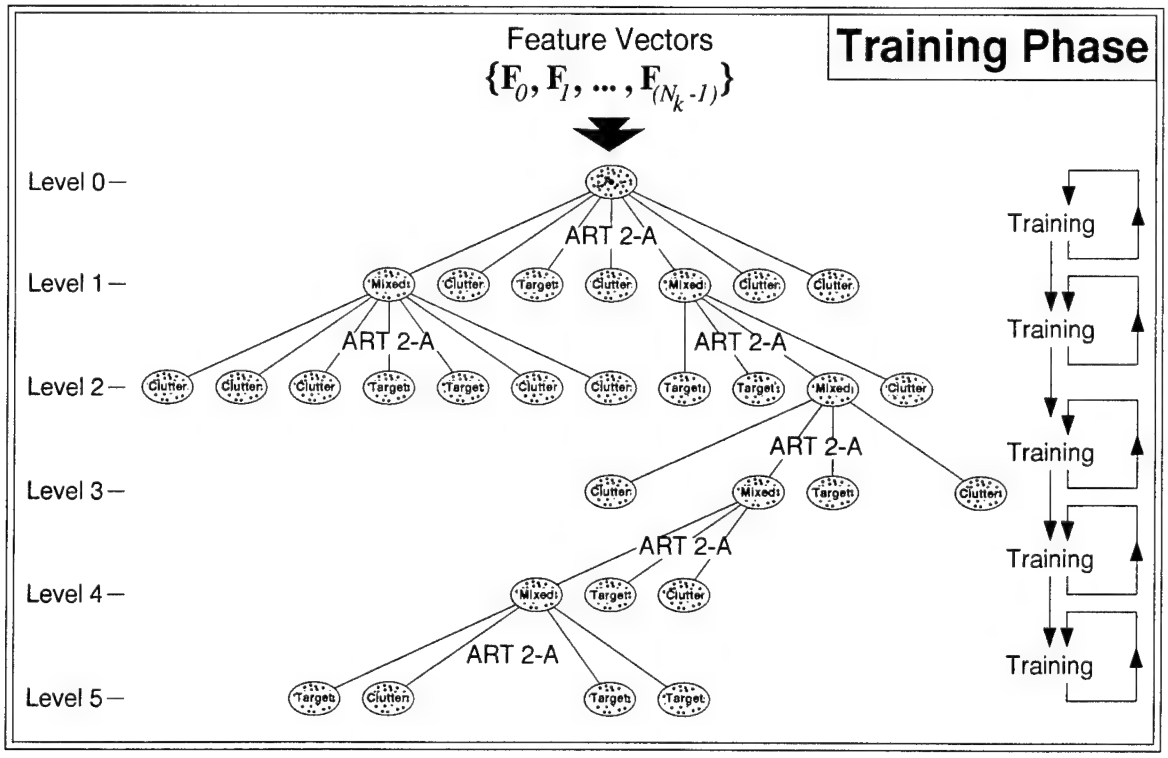

Figure 18. A typical tree structure may look like the above diagram. Maneuverability is accomplished by differencing the residual feature vector for a particular Level l, $\mathbf{F}_k^{(l)}$, with the winning cluster prototype, $\mathbf{z}_J^{*(l)}$. During training, **all** features are correctly classified by the system.

Hierarchical Classification: Test Phase. Once the training phase is completed, testing on an unknown set of vectors can be undertaken. Figure 19 describes the steps during this test process. As before, an example tree structure in Figure 20 will aid in the explanation of the algorithm.

Steps 1 and 2 in Figure 19 are similar to Figure 17. Thus, for a typical sensed (or unknown) feature vector, $\mathbf{F}_s$, the algorithm sets the index Level l to zero and stores the input feature vector into the current residual array, $\mathbf{F}_s^{(l)}$.

In Step 3, the ART 2-A network "processes through" with $\mathbf{F}_s^{(l)}$ by computing the maximum activation value, T_J, for a particular node at Level l. The maximum activation value, as defined in the previous section (see 3.2.1.1), is the best matching category found by the ART 2-A algorithm. Therefore, for node J and level l, the algorithm would find, $T_J^{(l)}$, such that

$$T_J^{(l)} = \max \left\{ T_j^{(l)} : j = 0, 1, \ldots, N^{(l)} - 1 \right\}.$$

Just as in the training version, Step 4 identifies (from $T_J^{(l)}$) the winning cluster prototype vector, $\mathbf{z}_J^{*(l)}$.

Step 5 is a logic statement to determine whether the prototype cluster is mixed. A cluster denoted as totally composed of target or clutter feature vectors will make the algorithm to go to Step 9 for classification. A mixed cluster, on the other hand, sends it to the next step (Step 6) to compute, $\mathbf{F}_s^{(l+1)}$. As shown in Figure 19, Step 7 increments l; while, Step 8 instructs the algorithm to return to Step 3 for the next set of T_j's in ART 2-A at the next level.

In Step 9, classification is performed by associating the cluster type with the residual feature vector, $\mathbf{F}_s^{(l)}$, at the final node destination.

Step 10 is a branching statement that sends the algorithm back to Step 1 for the next input feature vector.

In Figure 20, for the test version of the algorithm, the sensed feature vector, $\mathbf{F}_s$, is classified as a target vector on Level 5.

The following two comments can be stated from the above test version description of the hierarchical classifier.

1. Processing a feature vector through the tree structure occurs quickly (the algorithm maneuvers through the tree by computing the dot product: $T_j^{(l)} = \mathbf{F}_s^{(l)} \cdot \mathbf{z}_j^{*(l)}$). There's no time wasted in waiting for nodes to settle (as in the training algorithm).
2. The tree structure can "adapt" to new data by going back to a training mode to incorporate new training vectors (this capability was not added here; but can be an impetus for future work).

See Section 4.1.1 for performance results with the Demo C and ICD versions of the FLIR-based, background suppression algorithm.

1. Do **Steps 2 - 10** for sensed feature vector, $\mathbf{F}_s$.

2. Set Level $l = 0$ and $\mathbf{F}_s^{(l)} = \mathbf{F}_s$.

3. Process vector $\mathbf{F}_s^{(l)}$ through the ART 2-A network.

4. Identify winning cluster prototype, $\mathbf{z}_J^{*(l)}$.

5. If prototype cluster is not "mixed", *goto* **Step 9**.

6. Obtain residual feature vector,

$$\mathbf{F}_s^{(l+1)} = \left| \mathbf{F}_s^{(l)} - \mathbf{z}_J^{*(l)} \right|.$$

7. Set $l = l + 1$.

8. *Goto* **Step 3**.

9. Classify feature vector, $\mathbf{F}_s$, according to the class designation associated with winning prototype vector, $\mathbf{z}_J^{*(l)}$.

10. *Goto* **Step 1** for the next sensed feature vector; otherwise, **Stop**.

Figure 19. Once the tree structure is established after training, sensed feature vectors such as, $\mathbf{F}_s$, can be classified by the above algorithm. The method reduces the classification process to associating vector, $\mathbf{F}_s$, to the class of the nearest cluster or node in the tree.

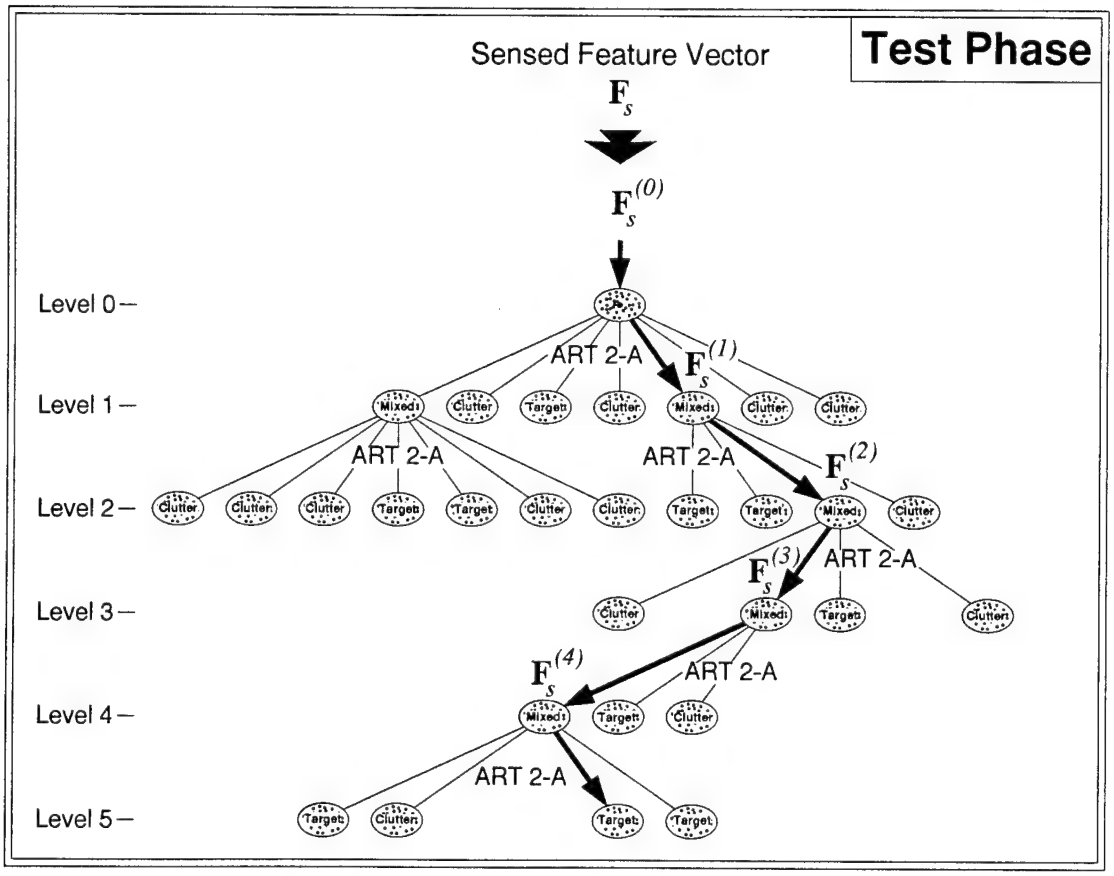

Figure 20. In this illustrative example, $\mathbf{F}_s$, is processed through the tree structure. The sensed feature vector is classified as a target vector at Level 5.

3.2.2 Classification in LADAR-Based Background Suppression

Classification techniques that were used for Rockwell's LADAR-based background suppressor is rooted in supervised neural networks: specifically, Radial Basis Functions (RBF's). RBF's are composed of a hidden (first) and output (second) layer. The hidden layer is made up of basis functions that produce a localized response to a input stimulus. Thus, they generate a nonzero response only when the input falls within a small localized region of input space (see references by Hush and Horne in [HH93] and Musavi *et al.* in [MACFH92] for lucid background material on radial basis functions). It is the same fundamental idea that is behind the coarse coding technique described in Section 3.1.1.3. Even though this approach is not part of the proposed ATD/I system (in the next section), it is a powerful method for classification and functional approximation applications.[16]

The implementation used here follows Hush and Horne's development in [HH93]. One begins with a Gaussian kernel function

$$u_{1,j} = exp\left[-\frac{\left(\mathbf{x}-\mathbf{w}_{1,j}\right)^T\left(\mathbf{x}-\mathbf{w}_{1,j}\right)}{2\sigma_j^2}\right]$$

for $j = 1, 2, \ldots, N_1$; where, $u_{1,j}$ is the output of the jth node for the first network layer, $\mathbf{x}$ is the input pattern from the 3D Hough approach in Section 3.1.2, $\mathbf{w}_{1,j}$ is the weight vector for the center of the Gaussian for node j, σ_j^2 is the normalization variable, and N_1 is the number of nodes in the first layer. Next, the output layer is

$$y_j = \mathbf{w}_{2,j}^T\mathbf{u}_1$$

for $j = 1, 2, \cdots, N_2$; where, y_j is the output of the jth node, $\mathbf{w}_{2,j}$ is the output weight vector, and $\mathbf{u}_1$ is the output vector from the first layer. In the classification mode, the neural network places the Gaussian kernel in the center of the data while modifying the circular decision boundary through training. The manner in which the decision boundaries are changed is via the normalization parameter, σ_j^2. Once the clustering algorithm is finished, a measure of the spread of the feature vectors is found for each node. The technique used here is the same as in [HH93]. Thus,

$$\sigma_j^2 = \frac{1}{M_j}\sum_{\mathbf{x}\in\Theta_j}\left(\mathbf{x}-\mathbf{w}_{1,j}\right)^T\left(\mathbf{x}-\mathbf{w}_{1,j}\right)\ ;$$

where; Θ_j is the (training) feature vector cluster center $\mathbf{w}_{1,j}$ and M_j is the number of feature vectors in Θ_j. In summary, learning is a two step process: parameters of the basis functions are first determined by the above equation, then followed by training in the output layer.

One can use many different learning algorithms for the two layers. Normally, learning in the hidden layer is accomplished with an unsupervised method. The unsupervised method is used mainly to generate clusters. In the background suppression implementation,

[16] Both RBF and ART 2-A based hierarchical classification (in Section 3.2.1.2) are fast algorithms, but the latter is faster because it can cope with larger databases while fitting nicely within a global control structure proposed in the enhanced ATD/I system of Section 3.2.3.

a streamline ART 2 neural network [CG87] developed by Thomas Ryan at SAIC [Rya88] called the Resonance Correlation Network (RCN) accomplished this task. The RCN software came from existing Rockwell developed code. For the output layer, a supervised learning algorithm is required. The Least Mean Squares (LMS) supervised classifier (similar to Hush and Horne in [HH93]) was developed for the task. RBF-based classification for background suppression is shown pictorially in Figures 21 and 22.

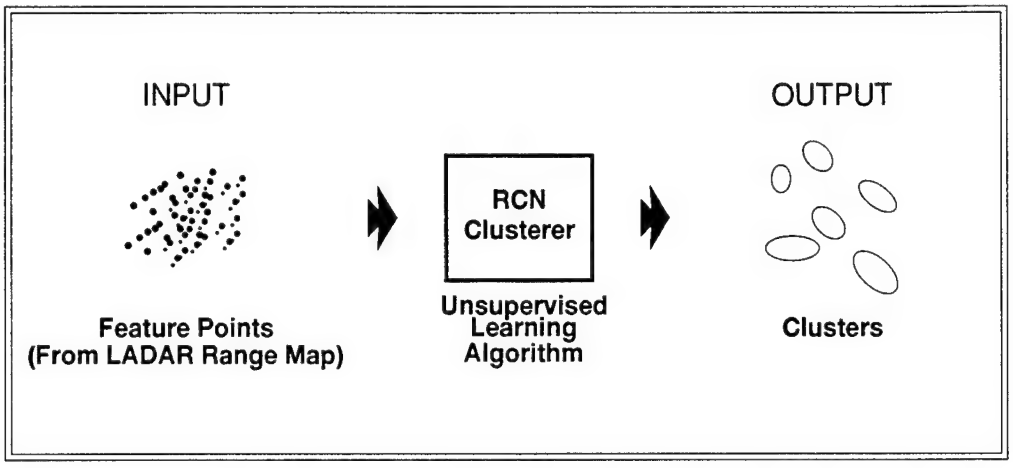

Figure 21. The LADAR-based, classification process implements a radial basis function neural network. The approach is a two-stage process: unsupervised clustering followed by supervised learning. In the above figure, the input feature vectors (processed through a 3D Hough based operation explained in Section 3.1.2) are input to a Resonance Correlation Network (RCN) for cluster generation.

The results using the radial basis function classifier with the 3D Hough feature extraction algorithms explained in Section 3.1.2 are given in Section 4.1.2.

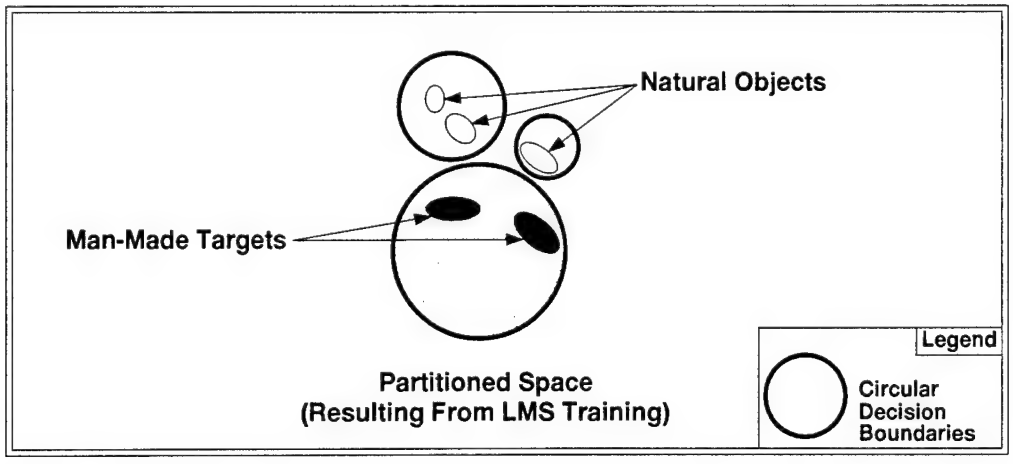

Figure 22. After clustering with the RCN neural network, the radial basis function approach generates decision boundaries through training using a standard Least Mean Squares (LMS) supervised learning algorithm. The goal is to separate man-made targets from natural objects.

3.2.3 Proposed Target Identification in Enhanced, FLIR/LADAR ATD/I System

In Section 3.1.3, the groundwork was established with an introductory description on the preprocessing and feature extraction portion of the proposed ATD/I system. In this section, the important, "back-end" target identification part will be discussed. To reiterate, the new target identification approach is based on employing techniques from lessons learned on the development of the FLIR-based and LADAR-based background suppression algorithms.

From the flow diagram of Figure 13, Figure 23 incorporates two new subsystems that pertain to the identification of targets: the on-line target recognition and off-line target learning algorithm suites. Beginning with the target learning subsystem, an off-line version of the aspect classifier that is similar to the background suppression recognition engine is shown in Figure 23. It receives 3D target models, FLIR/LADAR clutter models, and environmental information. The 3D target models are CAD prototypes with a pre-determined number of facets (in this application, they need not be to highly detailed; *e.g.*, 100 to 400 facets may be adequate). Boundary and plane data are extracted from the stored models in order to produce log-polar based feature vectors. In an analogous manner, the clutter model database operates on **known** clutter objects (from representative FLIR and LADAR imagery) according to the preprocessing and feature extraction subsystem (boxed area in Figure 23). The training vectors create the tree structure via the aspect classifier's learning process. It generates a type of aspect graph of the object viewpoints.[17] Specifically, the output viewpoint vector from the unknown object goes through a tree structure. The control mechanism of the algorithm manipulates the input vector through the tree. The process searches node by node in order to find the closest match between sensed and stored object representations. The manner in which aspect classification and hierarchical differencing would be performed is through the techniques of Section 3.2.[18] In feature space, the vectors representing the different targets may cluster according to similar object characteristics (*e.g.*, the majority of tank vectors may pass through turret clusters). The clusters are similar to what Waxman (in [WSBF93]) calls generic maps or objects. By learning clutter objects, the aspect classifier reduces the misclassification rate and increases overall system identification performance. Certainly, one can also feed known sensed target data during training and test phases. This capability is shown by the identified targets database in the off-line target learning system of Figure 23.

Environmental conditions are introduced into the ATD/I system in the form of heuristic rules. Environmental rules would modify the internal parameter settings in order to conform to changing terrain, time-of-day, and weather conditions. One example of

[17] An aspect graph representation of an object is a 2D plot of the different aspect categories. One can think of the nodes of the graph as depicting the object's viewpoints where connected lines of the plot represent allowed transitions between aspects.

[18] As mentioned in the Section 3.2 on hierarchical classification for background suppression, the aspect classification method is derived from the work done by Waxman and his co-workers at MIT Lincoln Laboratories (see references [WSBF93], [WS92], and [BW91]). One clear difference between Rockwell's method and the Lincoln Labs approach is that the internal object points (representing 3D plane information) are added to the overall feature vector for classification. In Waxman's approach, only boundary object points were used for recognition.

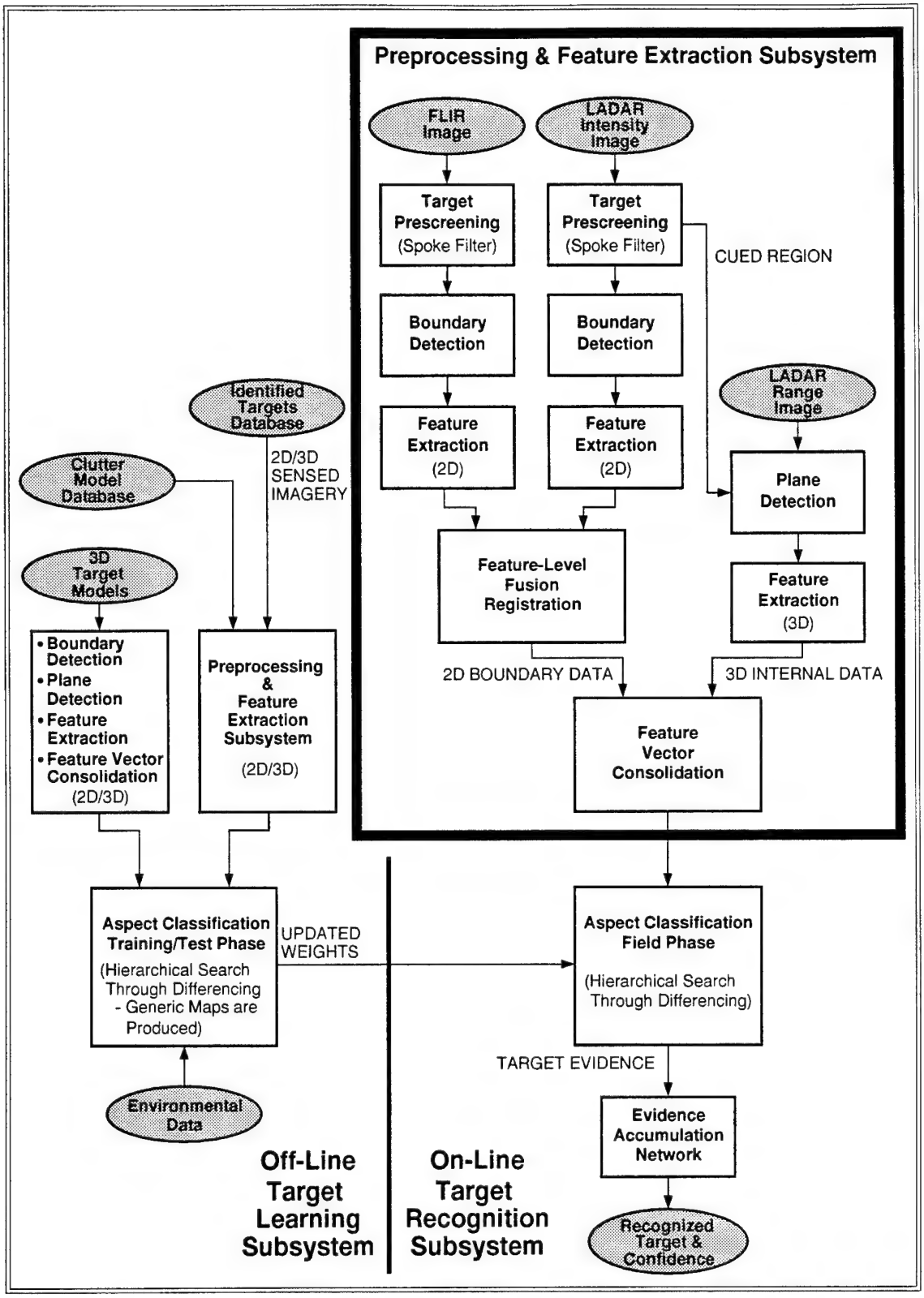

Figure 23. The proposed target identification approach is composed of three subsystems: preprocessing and feature extraction (described in Section 3.1.3), off-line target learning, and on-line target recognition algorithm suites.

such a rule could be to lower the vigilance parameter value in the ART 2-A networks for improved (classification) generalization. This action may be necessary because of decreased FLIR performance during the day; such as the time between 0900 to 1800 hours.

In the on-line target recognition subsystem, an overall feature vector originates from the preprocessing and feature extraction algorithm suite of Section 3.1.3 (see Figure 23). As mentioned in Section 3.1.3, the module appends 2D boundary and 3D internal data feature vectors to produce a global representation. The appended feature vector is fed into the (field phase) aspect classification module whose purpose is to classify the feature vectors according to the different object viewpoints. This module has been trained off-line with the most recent prototype, clutter, and identified target information. The new data is passed to the field version of the aspect classifier via updated weights.

Finally, the evidence accumulation network in Figure 23, would integrate over time the confidence values of the winning objects. The evidence network can be designed by using the evidential reasoning (Dempster/Shafer) method or Waxman's aspect network (see [WSBF93]) technique. In the latter approach, the network builds evidence according to permitted sequence of viewpoints during the training session. The aspect network self-organizes, similar to humans, in its learning mode the different aspect transitions of allowable target sequences. In the on-line mode, the process would build confidence over time for the identified target. Whether staring or scanning, confidence in a particular target would increase or decrease; but more importantly, the overall performance of the ATD/I system would improve.

In summary, an extensive effort was expended for planning and the future development of the enhanced ATD/I system. To maximize the effort and advance the state of the art on the new approach, the work was closely tied to a feature extraction/classification paradigm. This paradigm was applied first to the background suppression problem. Experience gained and selected software produced by the background suppression effort was to be carried over to the new ATD/I system.

4 Results from Relevant Data Collection Efforts

Portions of the enhanced FLIR/LADAR ATD/I system were developed during the course of the contract; mainly in the area of background suppression of FLIR and LADAR imagery. Section 4.1 documents results from the background suppression algorithms as it applies to relevant FLIR/LADAR databases. In Section 4.2, results are given that pertain to the application of the spoke filter on LADAR intensity imagery. **To use spoke filtering techniques on LADAR intensity imagery for registering FLIR (intensity) and LADAR (range map) data is a novel idea that originated from this contract.** The ultimate goal is to match targets using the two sensors. Fusing the two on a feature level is a fundamental step towards developing the proposed ATD/I system mentioned in Sections 3.1.3 and 3.2.3.

Also included in this section are variations between theory and practice. Obviously, discrepancies occur when one deals with real imagery. Not only algorithms are modified and changed when outputs are not what are expected; but more importantly, new insights into the problem come to mind during the process.

4.1 Background Suppression

Results obtained by background suppression are a reflection of the feature extraction/classification paradigm that was also to be the approach used for the enhanced ATD/I system. All tests and evaluations were processed with real image databases during the course of the contract. The results given in this section are in the form of

1) a confusion matrix for the FLIR background suppression algorithm delivered to Lockheed Martin in December, 1995,
2) tables describing percent correct/incorrect classification for test sets (after learning on training data) for the LADAR-based system, and
3) selected images explaining the differences between FLIR-based versions.

Unfortunately, because of limited resources no performance results were obtained with simulated data.

4.1.1 FLIR-Based Background Suppression

Beginning in the fall of 1994, the FLIR-only background suppression filter was primarily developed and evaluated for a year. It was trained and tested on FLIR imagery collected from two visits to the Demo C site in June, 1995. All visits took place at the Lockheed Martin facility in Denver, Colorado.

Section 4.1.1.1 contains a confusion matrix showing targets versus clutter results from imagery taken during the Denver visits. Results listed in Table 2 correspond to the output of the version given to Lockheed Martin for Demo II. Also in this section, as experience was gained with real imagery on the algorithm suite, a discussion is given on the different updates that were used during testing.

4.1.1.1 Performance Results from Demo C Site Visits

With a different set of weights from the Demo C version, Table 2 describes the performance results for the FLIR-based background suppression filter. The training database was the same for both: 31 targets and 36 clutter objects taken from 37 FLIR images.[19] The only difference between the two was in the off-line training curriculum. In the post Demo C case, it consisted in 67 vectors introduced to the classifier portion of the background filter in a random sequence of 67 for 10 cycles (rather than once as in Demo C). The results with this additional step was dramatic on the targets detected from the test set. As shown in Table 2, 88.9% detection was achieved (94.1% if one of the misses is discounted – see Table 2). The false-alarm rate is high at 33.5%. **It should be noted that the objective was to reduce the high number of detections while still retaining some clutter objects for further processing with back-end target identification algorithms. The algorithm suppresses false detections, but does not eliminate all false objects.** When the problem is placed in that framework, the high false-alarm rate may not be as bothersome.

Table 2. Performance results in medium to high clutter FLIR imagery demonstrates the target detection capability for the FLIR-based background suppression algorithm. Sixty-seven objects were selected manually from two June 1995 visits to the Denver Lockheed Martin facility. With random selection of the training vectors, the algorithm performed very well in detecting 16 out of 18 target-like objects (one of the misses was counted even though one vehicle obscured another; thereby changing the overall shape). Two-thirds of the clutter was classified correctly. False-alarms are not as costly as missing targets, since back-end target identification algorithms were able to reject clutter objects in many cases.

FLIR-BASED BACKGROUND SUPPRESSION FILTER TEST SET RESULTS
(CONFUSION MATRIX)

Object	Target	Clutter
	Number (% Correct)	Number (% Correct)
Target	16 (88.9)	2 (11.1)
Clutter	93 (33.5)	185 (66.5)

Training Set of 67 Objects Over 37 FLIR Images (size = 256 x 256):
31 Targets
36 Clutter

Test Set of 296 Objects Over 15 FLIR Images (size = 256 x 256):
18 Targets
278 Clutter

[19] The overall FLIR database was composed of 1) 37 training images obtained during the two June visits; and, 2) 15 test images gathered during the tech demonstration at Demo C. All objects were detected first by the spoke filter (67 for the training phase and 296 for the test exercise). The images were (256 × 256) pixel regions.

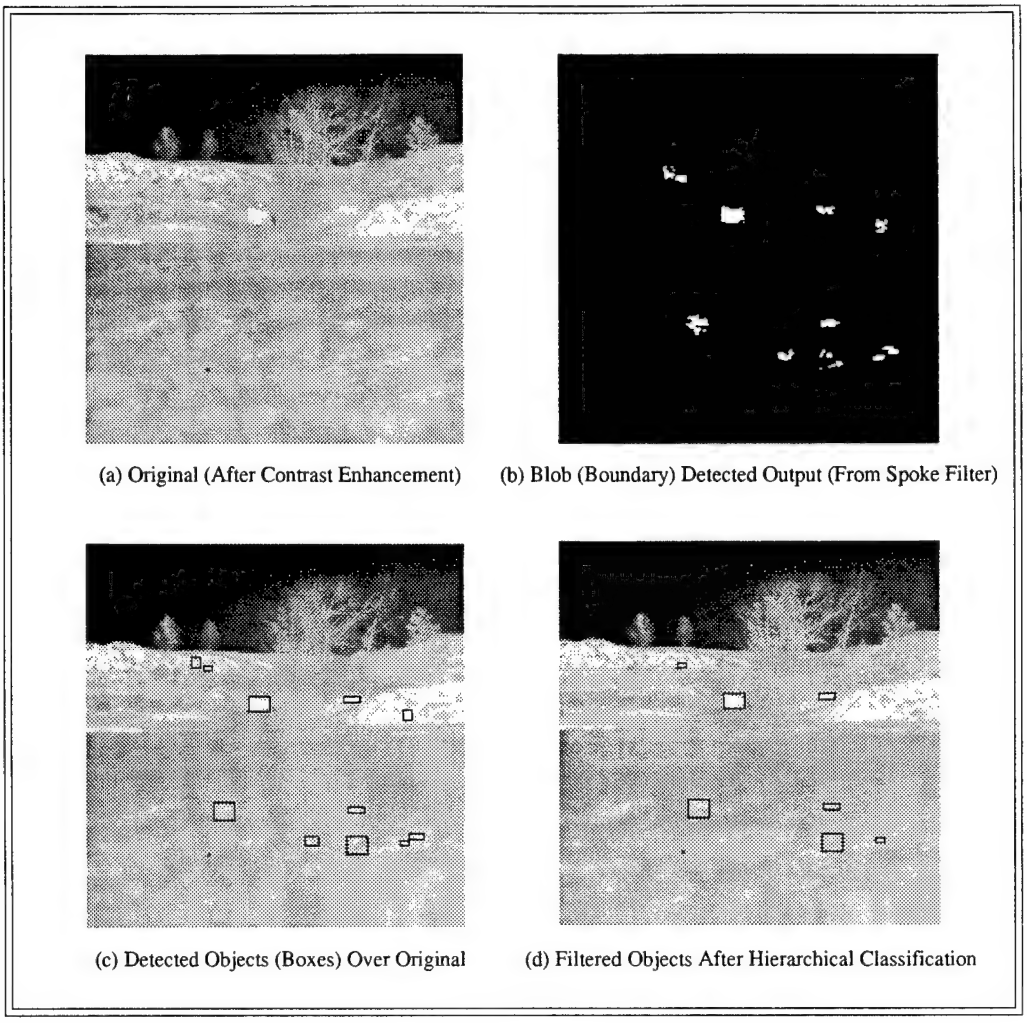

Figure 24. The FLIR-based background suppression version delivered to Lockheed Martin gave the results shown in Figure 24(d) for the FLIR image of Figure 8.

The algorithm suite delivered to Lockheed Martin in December 1995 consisted of a spoke filter (for blob detection), boundary detector (hot blobs only), log-polar feature extraction (with some minor but not catastrophic software errors), and two sets of trained weights (in the form of tree structures from the ART 2-A based, hierarchical classifier).[20]

Returning to the processed FLIR image in Figure 8, Figure 24 gives the output from the FLIR-based background suppression algorithm for the weights that produced Table 2. Notice that the hierarchical classifier only pruned off four clutter objects: going from 11 in Figure 24(c) to 7 in Figure 24(d). Comparing the images in Figure 24(b) and Figure 24(d), one sees that the algorithm kept a few that were obviously clutter, while the rest may be mistaken for targets. The confidence value for the true target turned

[20] The two sets consisted of the weights that produced the results for Table 2 and the output during Demo C.

out to be 0.8877, while the object in the upper left of Figure 24(d) had the next highest value at 0.8004.[21] From the boundary (blob) image in Figure 24(b), the upper left object looks like a small target.[22]

4.1.2 LADAR-Based Background Suppression

The algorithm was tested on LADAR imagery from the "hobby shop" database taken at Fort Carson, Colorado with the Alliant Techsystems LADAR (its resolution was approximately 1 pixel per foot on target; see reference [BHPHY94]). In order to evaluate the discrimination capability of the algorithm, seventy objects (35 target and 35 clutter) were selected manually from the LADAR image set. Through the feature generation portion of the algorithm, a 125-element vector was produced for each object. This process was an outcome of a variety of training/test scenarios performed on the feature set.

The results shown in Tables 3 through 6 summarized the work done during the period January to mid-July, 1994 (in preparation for Demo B). These tables demonstrate the loss of (classification) generality in the test set (and therefore recognition capability) as the number of clusters increase from two to fourteen. Notice that the unsupervised clusterer has an adjustable parameter that determines how closely feature vectors must match. A value near 0.3 will generate two or three clusters, while a number close to 1.0 can produce as many as 20 clusters for the feature vector set used in this test. Table 3 contains the classification results when the unsupervised clusterer (*i.e.*, the RCN algorithm) generates only two clusters. In this case, 85% correct classification is obtained for 20 test feature vectors. Table 4 shows 85% for three. Even up to six clusters, one can expect 80% correct classification for 20 unknown objects (after training on the remaining 50). Finally, Table 6 depicts 50% correct classification (or random chance) for the 20 test vectors. Table 6 demonstrates that a large number of clusters affect the algorithm to behave in a mere pattern memorization mode accompanied by poor generalization capability (especially when compared to results shown in Tables 3 and 4).

[21] The confidence measure used for the hierarchical classifier is normalized correlation (as the similarity criterion) between the input and cluster prototype vector associated with the winning node.

[22] This version of the algorithm does not incorporate range filtering; if included, all clutter objects would be removed except for the one to the right of the true target in Figure 24(d).

Table 3. Performance results for the LADAR-based background suppression algorithm with seventy objects selected manually from the Fort Carson "hobby shop" database. The radial basis function classifier categorized according to man-made or clutter classes after training on an unsupervised classifier that generated **two** clusters. The values inside the bold box show very good generalization results were obtained with this method.

LADAR-BASED BACKGROUND SUPPRESSION RESULTS FOR 2 CLUSTERS*				
TRAINING SET SIZE	TEST SET SIZE	TOTAL	PERCENT CORRECT (TEST SET)**	PERCENT INCORRECT (TEST SET)
65	5	70	**100.0**	0.0
60	10	70	**90.0**	10.0
50	20	70	**85.0**	15.0
40	30	70	**70.0**	30.0
30	40	70	**70.0**	30.0

* NUMBER OF CLUSTERS GENERATED BY THE UNSUPERVISED CLUSTERER IN THE FRONT-END PORTION OF THE RADIAL BASIS FUNCTION CLASSIFIER DURING TRAINING.

** CORRECTLY CLASSIFIED INTO MAN-MADE AND CLUTTER CATEGORIES.

Table 4. Performance results for the LADAR-based background suppression algorithm after the unsupervised classifier generated **three** clusters. The results obtained here approximate Table 3; *i.e.*, good generalization performance is achieved (see values inside bold box).

LADAR-BASED BACKGROUND SUPPRESSION RESULTS FOR 3 CLUSTERS*				
TRAINING SET SIZE	TEST SET SIZE	TOTAL	PERCENT CORRECT (TEST SET)**	PERCENT INCORRECT (TEST SET)
65	5	70	**100.0**	0.0
60	10	70	**80.0**	20.0
50	20	70	**85.0**	15.0
40	30	70	**70.0**	30.0
30	40	70	**70.0**	30.0

* NUMBER OF CLUSTERS GENERATED BY THE UNSUPERVISED CLUSTERER IN THE FRONT-END PORTION OF THE RADIAL BASIS FUNCTION CLASSIFIER DURING TRAINING.

** CORRECTLY CLASSIFIED INTO MAN-MADE AND CLUTTER CATEGORIES.

Table 5. Performance results for the LADAR-based background suppression algorithm after the unsupervised classifier generated **six** clusters. For this test set, the algorithm displays an obvious degraded performance when compared to the results in Table 3 and 4.

LADAR-BASED BACKGROUND SUPPRESSION RESULTS FOR 6 CLUSTERS*				
TRAINING SET SIZE	**TEST SET SIZE**	**TOTAL**	**PERCENT CORRECT (TEST SET)****	**PERCENT INCORRECT (TEST SET)**
65	5	70	100.0	0.0
60	10	70	70.0	30.0
50	20	70	80.0	20.0
40	30	70	70.0	30.0
30	40	70	72.5	27.5

* NUMBER OF CLUSTERS GENERATED BY THE UNSUPERVISED CLUSTERER IN THE FRONT-END PORTION OF THE RADIAL BASIS FUNCTION CLASSIFIER DURING TRAINING.

** CORRECTLY CLASSIFIED INTO MAN-MADE AND CLUTTER CATEGORIES.

Table 6. Performance results for the LADAR-based background suppression algorithm after the unsupervised classifier generated **fourteen** clusters. Here, substantial degradation has taken place. Mere pattern memorization with very little classification generalization has occurred.

LADAR-BASED BACKGROUND SUPPRESSION RESULTS FOR 14 CLUSTERS*				
TRAINING SET SIZE	**TEST SET SIZE**	**TOTAL**	**CORRECT (TEST SET)****	**PERCENT INCORRECT (TEST SET)**
65	5	70	90.0	10.0
60	10	70	60.0	40.0
50	20	70	50.0	50.0
40	30	70	56.7	43.3
30	40	70	72.5	27.5

* NUMBER OF CLUSTERS GENERATED BY THE UNSUPERVISED CLUSTERER IN THE FRONT-END PORTION OF THE RADIAL BASIS FUNCTION CLASSIFIER DURING TRAINING.

** CORRECTLY CLASSIFIED INTO MAN-MADE AND CLUTTER CATEGORIES.

4.2 FLIR/LADAR Fusion for Proposed ATD/I System

Due to program restrictions, there was not enough time to do extensive work on the proposed ATD/I system (described in Sections 3.1.3 and 3.2.3); although, one important experiment was accomplished. The idea was to use the FLIR-based background suppression filter of Section 3.1.1 on LADAR intensity imagery. **Can target detection be possible with a background suppression algorithm that was primarily developed for FLIR imagery? The answer should be yes; since, the manner in which the filter operates, by accumulating digital blob evidence, makes it (imaging) sensor independent.** As demonstrated in the next section, the results with LADAR intensity images show that such an approach works surprisingly well. With the right adjustments on the FLIR-based background suppression filter, it can detect and segment targets from background just as well, sometimes better, than comparable scenes with 3-5 μm Amber FLIR imagery.

4.2.1 Selected Imagery – LOCAAS MICOM Collection

The FLIR-based background suppression filter was tested on three (representative) LADAR intensity images (containing five targets) from the LOCAAS Micom collection. As this section will point out, the output produced by the algorithm indicates that it performed soundly on the LADAR intensity images. With more work and testing, a modified version of this algorithm can be included in the development of the feature extraction portion of the proposed, FLIR/LADAR ATD/I system shown in Figure 13.

Testing was performed on the LOCAAS 2-channel LADAR MICOM tower data. It was collected by Lockheed Martin Vought Systems (formerly, Loral Vought Systems). The database, 40.2 megabytes in size, contains intensity and range (diode pumped) LADAR imagery. The LOCAAS collection includes a variety of targets (*e.g.*, M60, M113, T72, 5 ton truck, personal car, water tower, trees, *etc.*). In April 1996, this imagery was placed on the Lockheed Martin file server for use by the RSTA community.

The background suppression software program employed the trained weight set (in tree structure form) that was used for Demo C (July, 1995). Obviously, several changes were made in order to accommodate the LADAR imagery. The following two major changes were made to the FLIR-based algorithm.

- The boundary portion of the algorithm (see the subsection on spoke filtering techniques in Section 3.1.1.1) incorporated "black-hot" software for darker-than-background targets (this feature is required for any future FLIR-based versions too). LADAR intensity imagery may contain targets that are either darker or brighter than the background.
- The radius size for digital blobs that can be detected with the spoke filter went from 10 to 17 pixels. This parameter governs the approximate target size screening capability of the spoke filter module of the background suppression algorithm. The LOCAAS targets are at a closer range (*i.e.*, between 100 and 300 meters) than the FLIR databases used in the development of the original algorithm.

Also, in order to go from an image size of (340×148) pixels to (256×256) required for the suppression filter, the selected LADAR intensity imagery were cropped and padded with a rough estimate of the gray-level background (*i.e.*, grey level value of 155). These

changes were all that could be done with the limited amount of time.

The results that are shown in Table 7 were surprisingly good for the three LOCAAS intensity images. The spoke filter part of the background suppression algorithm detected all five targets in the image set. The back-end hierarchical classifier detected three out of five targets. In classification terminology, three were classified as targets (in image `f700561i.dat`, the M60A3 tank two hits were classified as one target), two incorrectly classified as clutter objects, and 24 objects correctly classified as clutter. From the small image set used for this experiment, the results compare favorably to the very best of the 3-5 μm Amber FLIR imagery taken for the RSTA community.

Table 7. Results show that the FLIR-based background suppression filter is capable of detecting and segmenting targets from LADAR intensity imagery. More work is needed for feature level fusion; but, the path taken here looks very promising.

FLIR-BASED BACKGROUND SUPPRESSION FILTER RESULTS ON SELECTED LOCAAS LADAR INTENSITY IMAGERY

FILE NAME	TYPE	SUBTYPE	RANGE (Meters)	SPOKE (% OF TARGET) DETECTION	CLASSIFIER (% OF TARGET) DETECTION
f7004a1i.dat*	Tank	T72	253.20	Yes (95)	No (0)
f7004a1i.dat	APC	BMP	273.75	Yes (100)	Yes (100)
f700561i.dat**	Tank	M60A3	237.10	Yes (Approx. 15)	Yes (Approx. 10)
f7004e1i.dat***	Truck	M35	295.20	Yes (100)	No (0)
f7004e1i.dat	Tank	M48	317.25	Yes (100)	Yes (100)

*SPOKE FILTER: 3 DETECTIONS AND 2 FALSE ALARMS; CLASSIFIER: 4 CORRECTLY CLASSIFIED (1 TARGET & 3 CLUTTER) OUT OF 5.

**SPOKE FILTER: 3 DETECTONS ON SAME TARGET (GUN BARREL, FRONT PART OF TANK, & 2 BACK WHEELS); CLASSIFIER: 14 CORRECTLY CLASSIFIED (2 TARGET & 12 CLUTTER) OUT OF 16.

***SPOKE FILTER: 2 DETECTIONS AND 7 FALSE ALARMS; CLASSIFIER: 8 CORRECTLY CLASSIFIED (1 TARGET & 7 CLUTTER) OUT OF 9.

5 Conclusions

Starting from September 12, 1993, the objective for this program was to develop a high performance FLIR/LADAR sensor fusion algorithm suite for target identification that advances the state of the art in image understanding within a SSV/RSTA environment. During the last 36 months, as shown in Figure 25, the task has emphasized background suppression in order to achieve target detection and identification with FLIR and LADAR sensor data. The justification for stressing background suppression is that one must be willing to tackle clutter rejection at the beginning of target detection and identification development. It cannot be an afterthought or an "add on" after the system is almost completed. It must be integrated into any planned ATD/R/I system. Ground rules must be established on the difference between targets and clutter objects. To obtain high detection, low false-alarm rates, and robust identification of targets, one must deal with background suppression and multi-sensor fusion on a feature level at the onset.

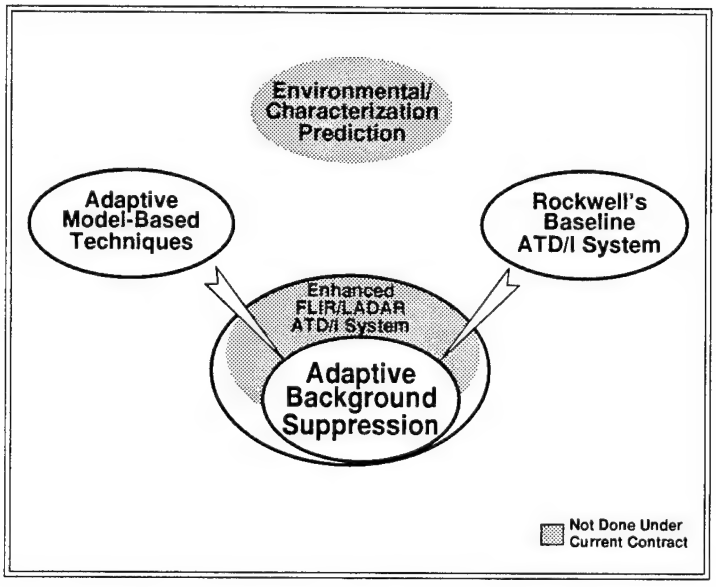

Figure 25. Background suppression techniques, whether FLIR or LADAR oriented, were used towards the target detection and identification objective.

The approach focused on two important ideas to achieve significant improvement in target detection and identification capability: 1) to characterize background clutter in FLIR and LADAR imagery by formulating the problem in a feature extraction/classification paradigm; and, 2) to use the background suppression approach as the basis for building an advanced ATD/I system.

On the characterization of background clutter, by employing the feature extraction/classification techniques described in Section 3, it makes the overall task of target detection and identification easier. One can introduce learning techniques into the problem; therefore, making the system more adaptive to a changing environment. It is a "smart" prescreener for the back-end target identification subsystem. If the background suppression algorithms identify objects that are obviously clutter, it then makes the job of

accepting only targets and few very near target-like clutter objects more manageable. Detecting and identifying clutter removes potential target candidates before further (costly) processing is performed by the ATD/I system. The implication by the methodology done here is that it is better than having the ATD/I system process (or more specifically classify/identify) all likely candidates through the system. The justification for suppressing clutter objects is necessary because lowering the target detection threshold in sensed imagery to detect faint or hard-to-see targets increases the false-alarm rate. A higher number of clutter objects intermixed with real targets translates into a lower performance for any established (or proposed) ATD/I system.[23] Lower performance is virtually guaranteed because the system spends longer time evaluating and potentially misclassifying spurious background clutter. **The background suppression techniques developed here aid in target detection and identification by pruning much more effectively false clutter off target candidate lists.**

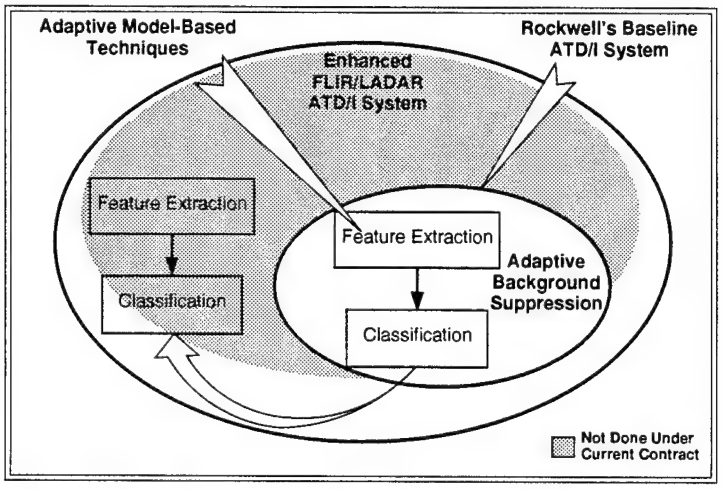

Figure 26. The approach implemented for the enhanced FLIR/LADAR ATD/I system was the feature extraction/classification paradigm.

On the issue of building a more powerful ATD/I system, the approach was to use the platform implemented for background suppression as the foundational algorithm suite – see Figure 26. As stated in Sections 3.1.3 and 3.2.3, an innovative feature extraction/classification paradigm that would eventually incorporate FLIR and LADAR data was first applied to FLIR-based and LADAR-based background suppression. The intent was to gain developmental experience on the new techniques. The background suppression feature extraction/classification algorithms would then be modified and extended to target identification using a model-based (CAD) approach as described in Sections 3.1.3 and 3.2.3. **The new system was to have fused data on a feature rather than pixel level basis. Faster registration of objects contributing to a higher level of synergy between FLIR and LADAR sensors was the goal here (see the overall flow diagram in**

[23] It does not mean that the system should strive for zero percent false-alarms on the target detection portion – which is a mistake that some people make. In many ATD/I systems, to contend for a zero false-alarm rate is too costly. This goal places too much responsibility on the target detection subsystem. A much better approach is to have the target identifier cope with an adequate number of targets that includes some obvious false-alarms, but not too high so as to overwhelm the ATD/I system. This rationale was used in the FLIR and LADAR background suppression algorithms developed under the contract.

Figure 23). The new target identification system will enhance mission effectiveness by: 1) a higher detection and lower false-alarm rate that is due to characterization of clutter on the front-end of the ATD/I system; 2) fusing FLIR and LADAR data with a common feature vector that will create a robust environment for accurate classification and identification; and, 3) a hierarchical classification that will maneuver through feature space adaptively for a higher rate of target identification performance. Such a ATD/I system will lay the foundation for future handling of target articulation.

6 Legacy

Rockwell brought several important products to the UGV/RSTA program. The critical item of the project was the work done towards improving target detection and identification. This work was in the form of suggestions and flow diagrams mentioned in Sections 3.1.3 and 3.2.3 of the proposed, FLIR/LADAR ATD/I system. The method used towards the goal of advancing the state of the art was the developing and testing of ATD/I algorithms on real FLIR and LADAR imagery. These algorithms were mainly in the area of background suppression. The rule of thumb that was followed in developing a new ATR system was that any advanced target detector or identifier that does not incorporate background or clutter suppression from the ground up is doomed to poor performance in most realistic scenarios. Many hours of thought, planning, and technical experience were put into the enhanced system work which culminated in the present report.

Second in importance was the development of the FLIR-based background suppression software module. The module was delivered to Lockheed Martin (Denver) in December, 1995. The aim was to integrate it on to the SSV's as a FLIR stationary detector for Demo II in May, 1996 at Fort Hood. But for some unknown reason it was not used by Lockheed Martin for the demonstration.

The following sections describe in more detail the Rockwell, FLIR/LADAR fusion target detection and identification legacy of products. It is partitioned into: 1) the major technical reports and delivered software in Section 6.1; and, 2) suggestions on a future direction for RSTA and general recommendations in Section 6.2.

6.1 Technical Reports and Software

During the life of this contract's period of performance, several technical reports and software programs were produced by the effort. The following items describe major products that were finished during the 36 month period.

- **Final Report:** From this report enough documentation is given that allows someone downstream to understand and build on Rockwell's contribution. It explains the motivation and rationale behind the approach to improve ATR systems in general. It comments on why state of the art techniques such as hierarchical neural nets and generic maps can be incorporated in a ATD/I system in order to perform multi-sensor fusion, background clutter suppression, and multi-frame recognition. Also, documented results are presented on how well the background suppression filter (both FLIR and LADAR) performed with real imagery.
- **FLIR-Based Background Suppression Software:** A software package was developed and delivered to Lockheed Martin in Denver during the month of December, 1995. It incorporated many of the FLIR-based background suppression ideas in Section 3.1.1. As mentioned in the performance results section (see Section 4.1.1), it was primarily trained and tested on FLIR imagery collected from two visits to the Demo C site in June, 1995. The programs were written in Kernighan and Ritchie "C" source code; they conformed to Lockheed Martin's

Interface Control Document guideline for real-time testing (on an SSV). The algorithm suite consisted of a spoke filter (for blob detection), boundary detector (hot blobs only), log-polar feature extraction (with some software errors but not catastrophic), and the trained weights (tree structure from an ART 2-A based, hierarchical classifier) from the same database used to obtain the performance results shown in Table 2.

- **Interim Technical Report:** On April 15, 1994, a 13 page interim technical report was sent to DARPA depicting the progress made during the first six months of the contract (see [Roc94] for the primary reference and [GW94] for a paper summarizing most of the work done in 1994). Besides describing a preliminary form of the enhanced ATD/I system given here, the document contained valuable information concerning performance results and a general flow diagram of Rockwell's baseline (model-based), FLIR/LADAR ATR system. This system was developed under previous ATR contracts and in-house IR & D. In a lab demonstration for Demo B (July, 1994), it correctly identified an APC and M-60 tank from Fort Carson "hobby shop" database. Finally, the report made a suggestion on incorporating micro-Doppler (vibration) in any future ATR systems. Using micro-Doppler to identify targets may answer some of the concerns with the RSTA algorithms. For example, one problem is the effectiveness of target detection and identification over realistic distances (4 to 5 kilometers minimum). The issue was brought up during the Demo II concluding workshop at Killeen, Texas (June, 1996). This problem and others are addressed in the next section.

The final section of this report will focus on important issues to solve, lessons learned on this program, and recommendations relating to any "follow on" RSTA work. The motivation is to satisfy the military user. And, in the course of satisfying the customer one cannot help to advance the state of the art in image understanding and ATR technology.

6.2 Future Direction and Recommendations

The attempt by the RSTA co-contractors is a noble one: to push the envelope in image understanding and ATR technology while satisfying the military customer. But to be realistic is to know the technical limitations of the product that one offers to the user; and, to say what can or cannot be done. After a reality check, one should next lay out a plan to reach achievable goals. This section deals with answering some of the military criticism with RSTA and technical hurdles necessary to arrive at useful ATD/R/I systems.

In answering criticism and recommending new directions, the problem will first be stated, then followed by a response. In replying to the problem, new areas of potential research will be pointed out whenever possible. As noted, some of the tough questions where brought out by military users at the Demo II workshop. Rockwell representatives attended the Demo II concluding workshop during June 19-20, 1996 in Killeen, Texas. The following problems bring out some of the main concerns with the RSTA algorithms; and indirectly, pave the way for new directions that will ultimately improve the technology.

- **Let's be practical, target detection, recognition, and identification should be used over 5 km (maybe, 4km minimum); anything closer would not be**

realistic. One military officer at the Demo II concluding workshop made the above statement (or words to the effect). He touched on a very important concern to the military. The 4-5 km distance seems be the threshold for many military scenarios.

Response: At a 4-5 km distance, most first generation FLIR's would have approximately one to two pixels-on-target (POT) if any. Second or third generation FLIR's may be somewhat better. At minimum, for example, a FLIR-based blob detector would require between 5 to 10 POT's for the smallest targets. A new approach would be to use "track-before-detect" techniques followed by micro-Doppler.

New Direction: Track-before-detect algorithms detect targets of known characteristics over a image sequence. Usually, for example, targets may have a low observable cross section at a far distance. One may utilize knowledge of the object's dynamic behavior in order to detect it (*e.g.*, M1 tank's speed, maneuverability, and potential paths for a certain terrain). Detection is possible by looking within a spread of kinematically possible templates. For example, Seidman in [Sei90], has the idea of utilizing normally unused information remaining in the pattern formed by a track in order to dig out tracks from a highly noisy background. One can increase the sensor's effectiveness by utilizing a neural network to recognize these "hidden by the noise" patterns. A RSTA implementation of track-before-detect algorithms would be to scan with a FLIR in some predetermined fashion over a desired area looking for targets with characteristically kinematic pattern (similar to Seidman). One would repeatedly do this scanning over a period of time in order to build up target and/or clutter evidence. Algorithms (*e.g.*, motion-based, blob-based, *etc.*) would be specifically designed to gather such evidence over time. If one finds suspected objects (or regions), then a micro-Doppler approach can be used for target identification. Coherent LADAR sensors can measure target Doppler. They allow for the determination of target velocity (useful for detection) and vibration (useful for identification). Vibration-based target identification requires a classification algorithm suite and target vibration signatures.

- **The FLIR's on the SSV's just don't do the job; especially, during daylight hours. What we need are second or third generation FLIR's.** The FLIR's on the SSV's are currently first generation (*i.e.*, 3-5 μm Amber FLIR). Performance is poor during daylight hours (say, 0900 to 1800 hours on a typical sunny day). For example, rocks, shrubs, trees, buildings, and roads become just as hot as targets; they work against finding adequate object boundaries.

  Response: Second (and possibly third) generation FLIR are not much better. Also, second generation FLIR's are an order of magnitude more costly than the FLIR's used on the SSV's. Cost would be prohibitive at this

point. A different combination of sensors and algorithms may give higher performance for ATR applications. In particular, the work done by the RSTA contractors, Colorado State and Johns Hopkins, looks promising. Augmenting FLIR with color-based recognition and visual polarization techniques may be a fruitful path.

New Direction: A new approach, again, would be to combine the first generation FLIR with a low-cost LADAR micro-Doppler unit. First, scan areas of interest with the FLIR, then direct the LADAR to suspected targets. The unit would be fast, pruning off obvious clutter candidates quickly. It would able to work in conjunction with more sophisticated ATD/R/I systems.

- **It seems that some are shy about using active sensors for target detection and identification. There are military scenarios where one could use them with no problem.** The same military officer who made a comment on realistic images of RSTA sensors at the Demo II workshop made another important point concerning active sensors. He stated that even activities as benign as looking through a pair of binoculars can be detected by the enemy, if the sun's glare hits them at a certain angle. The inference here is that an active sensor such as LADAR can be used by the military customer for appropriate applications.

  Response: The military's thinking on this issue is almost completely turned around from just a few years ago. It would be to the RSTA community's advantage to incorporate more active sensors for imaging, detection, and ranging applications.

  New Direction: Low-cost LADAR's and MMW units can be integrated into any sensor suite. A scheme similar to what is described in this report for the enhanced, ATD/I system (in Sections 3.1.3 and 3.2.3) would improve performance. An added feature would be to do stored model verification on the target identified by the ATD/I system. This technique would implement edge/line detection (see Canny [Can86]) and matching algorithms (*e.g*, Bejanin [BHMN94]) in order to back project the chosen wire-frame model to the sensed object (at the estimated orientation and position) a la Lowe's method in [Low85] or the University of Southern California's approach in reference [BHMN94].

- **Can anything be done to exploit a sequence of images in future target detection and identification systems?** At the present time, moving target detection (obviously) is the only RSTA area that does anything with a previous history of images. Why are RSTA co-contractors not taking full advantage of multi-frame imagery for (stationary) target detection and identification?

  Response: Much work is required to implement a target detection and identification system that takes into account previous target history in the current scene for real-time use. Blending previous target history to the present target hypothesis is difficult. It is basically an art form at this stage of ATR technology.

New Direction: Future ATR work must face the issue of sequential target identification. New research should center on not only fusing complementary sensor suites; but, just as important, previous target history. **Resetting the ATR system for every new image fails to capitalize on what transpired before.** To implement Waxman's approach (in [WSBF93]), which was briefly mentioned in this report in Section 3.2.3, may be a good start. The idea is elegant (stemming from visual biological studies with primates). One creates different aspect transitions off-line (analogous to a transition matrix in controls theory). These transitions depict different target sequences that would be permitted by the network (*e.g.*, a front part of a tank would not be immediately followed by its back side). Once the learning network accepts only permitted target sequences, then in the on-line mode it would score the real-time target sequence accordingly. The network would introduce a confidence value on the recognized target aspect, disallowing wildly discordant views in the process. Thus, previous target history would have a contributory part in identifying the target for the current scene.

In summary, new ATR systems must fuse passive and active sensors (automatically registering them either on a pixel, feature, or symbolic level), deal with articulation and occlusion, and integrate past imagery as it works towards target detection, recognition, and identification in real-time. But, whatever approach to the problem is taken, it must not lose sight of the end-user. For example, at the concluding Demo II workshop in Killeen (Texas), one of the Army's technicians, trained specifically on the UGV/SSV's beforehand, claimed that the FLIR stationary target detector was disappointing. He said that there were too many detections (too many false-alarms). Detecting targets during the day over varying environmental conditions with a FLIR is a difficult but a fundamental problem. Therefore, let us work to advance the state of the art with sound, fundamental image processing practices as the community strives to develop more advanced image understanding algorithms. The work described in this report attempted to do both.

Bibliography

[AIGL90] Jose Ambros-Ingerson, Richard Granger, and Gary Lynch. Simulation of palecortex performs hierarchical clustering. *Science*, 247:1344–1348, 16 March 1990.

[BGW95] H. Borsi, E. Gockenbach, and D. Wenzel. Separation of partial discharges from pulse-shaped noise signals with the help of neural networks. In *IEE Proceedings - Science, Measurement, and Technology*, volume 142, pages 69–74, January 1995.

[BHMN94] M. Bejanin, A. Huertas, G. Medioni, and R. Nevatia. Model validation for change detection. In *1994 Image Understanding Workshop Proceedings - ARPA*, volume 1, pages 287–294, November 1994.

[BHPHY94] J. Ross Beveridge, Steve Hennessy, Durga P. Panda, Bill Hoff, and Ted Yachik. November 1993 Fort Carson RSTA data collection - final report. Technical report, Colorado State University, 19 January 1994.

[BW91] A. A. Baloch and A. M. Waxman. Visual learning, adaptive expectations, and behavioral conditioning of the mobile robot MAVIN. *Neural Networks*, 4(3):271–302, 1991.

[Can86] John F. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, November 1986.

[CG87] Gail A. Carpenter and S. Grossberg. ART 2: Self-organization of stable category recognition codes for analog input patterns. *Applied Optics*, 26(23):4919–4930, December 1987.

[CGR91] Gail A. Carpenter, S. Grossberg, and George Rosen. ART 2-A: An adaptive resonance algorithm for rapid category learning and recognition. *Neural Networks*, I(4):493–504, 1991.

[FKMHH92] R. J. Fogler, M. W. Koch, M. M. Moya, L. D. Hostetler, and D. R. Hush. Feature discovery via neural networks for object recognition in SAR imagery. In *1992 IEEE International Joint Conference on Neural Networks*, volume 4, pages 408–413, 1992.

[GW94] Victor M. Gonzalez and Paul K. Williams. Summary of progress in FLIR/LADAR fusion for target identification at Rockwell. In *1994 Image Understanding Workshop Proceedings - ARPA*, volume 1, pages 495–499, November 1994.

[HH93] D. R. Hush and B. G. Horne. Progress in supervised neural networks. *IEEE Signal Processing Magazine*, pages 8–39, January 1993.

[HSK95] Tetsuji Haga, Koichi Sasakawa, and Shinichi Kuroda. The detection of lane boundary markings using the modified spoke filter. In *IEEE 1995 Symposium Proceedings of*

the Intelligent Vehicles, pages 293–298, September 1995.

[KC89] R. Krishnapuram and D. Casasent. Determination of three-dimensional object location and orientation from range images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(11):1158–1167, November 1989.

[Low85] David G. Lowe. Perceptual organization and visual recognition. Kluwer Academic Publishers, Inc., Boston, 1985.

[MACFH92] M. T. Musavi, W. Ahmed, K. H. Chan, K. B. Faris, and D. M. Hummels. On the training of radial basis function classifiers. *Neural Networks*, 5(4):595–603, 1992.

[MS81] L. G. Minor and J. Sklansky. The detection and segmentation of blobs in infrared images. *IEEE Transactions on Systems, Man, and Cybernetics*, 11(3):194–201, March 1981.

[RM86] David E. Rumelhart and James L. McClelland. Parallel distributed processing, volume 1. MIT Press, Cambridge, MA, 1986.

[Roc94] Rockwell. FLIR/LADAR fusion for target identification - April 15, 1994. Technical report, Rockwell International Corporation, C94-14.1/034, 15 April 1994.

[Rya88] T. W. Ryan. The resonance correlation network. In *1988 IEEE International Joint Conference on Neural Networks*, volume 1, pages 673–680, 1988.

[Sei90] A. N. Seidman. Neural networks and digital avionics. In *IEEE/AIAA/NASA 9th Digital Avionics Systems Conference Proceedings*, pages 669–678, October 1990.

[TG94] Ian Taylor and Mike Greenhough. S-ART: a modified ART 2-A algorithm with rapid intermediate learning capabilities. In *1994 IEEE International Joint Conference on Neural Networks*, volume 2, pages 606–611, 1994.

[WS92] A. M. Waxman and M. Seibert. Adaptive 3D object recognition from multiple views. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):107–124, February 1992.

[WSBF93] A. M. Waxman, M. Seibert, A. M. Bernardon, and D. A. Fay. Neural systems for automatic target learning and recognition. *The Lincoln Laboratory Journal*, 6(1):77–116, 1993.